CYBERSECURITY SURVIVAL GUIDE

Fundamental Principles and Best Practices

Sixth Edition | April 2022

Lawrence C. Miller, CISSP

Cybersecurity Academy

Table of Contents

Module 1 – Fundamentals of Cybersecurity 1

1.0 Cybersecurity Landscape 1

1.0.1 Modern computing trends 2

1.0.2 New application threat vectors 10

1.0.3 Turbulence in the cloud 12

1.0.4 SaaS application risks 15

1.0.5 Compliance and security are not the same 18

1.0.6 Recent high-profile cyberattack examples 21

1.1 Cyberthreats 25

1.1.1 Attacker profiles and motivations 25

1.1.2 Modern cyberattack strategy 28

1.1.3 MITRE ATT&CK framework 32

1.2 Cyberattack Techniques and Types 35

1.2.1 Malware and ransomware 35

1.2.2 Vulnerabilities and exploits 41

1.2.3 Business email compromise (BEC) 43

1.2.4 Bots and botnets 45

1.2.4.1 Spamming botnets 47

1.2.4.2 Distributed denial-of-service botnets 48

1.2.4.3 Financial botnets 49

1.2.5 Advanced persistent threats 49

1.2.6 Wi-Fi attacks 52

1.2.6.1 Wired Equivalent Privacy 52

1.2.6.2 Wi-Fi Protected Access (WPA/WPA2/WPA3) 52

1.2.6.3 Evil Twin 56

1.2.6.4 Jasager 57

1.2.6.5 SSLstrip 58

1.2.6.6 Emotet 59

1.2.6.7 Doppelganger 60

1.2.6.8 Cookie guzzler 60
1.3 Network Security Models 61
1.3.1 Perimeter-based network security strategy 61
1.3.2 Zero Trust security 63
1.3.2.1 Core Zero Trust design principles 64
1.3.2.2 Zero Trust conceptual architecture 65
1.3.2.3 Key Zero Trust criteria and capabilities 69
1.3.2.4 Implementing a Zero Trust design 70
1.4 Security Operating Platform 72
Module 2 – Fundamentals of Network Security 78
2.0 The Connected Globe 78
2.0.1 The NET: How things connect 79
2.0.2 Introduction to networking devices 79
2.0.3 Area networks and topologies 80
2.0.4 Domain Name System 86
2.0.5 The internet of things 88
2.1 Physical, Logical, and Virtual Addressing 94
2.1.1 IP addressing basics 98
2.1.2 Introduction to subnetting 101
2.2 Packet Encapsulation and Lifecycle 104
2.2.1 The OSI and TCP/IP models 105
2.2.2 Data encapsulation 111
2.2.3 Routed and routing protocols 112
2.3 Network Security Technologies 116
2.3.1 Firewalls 116
2.3.1.1 Packet filtering firewalls 116
2.3.1.2 Stateful packet inspection firewalls 116
2.3.1.3 Application firewalls 117
2.3.2 Intrusion detection and intrusion prevention systems 117
2.3.3 Web content filters 118

2.3.4 Virtual private networks 119
2.3.4.1 Point-to-Point Tunneling Protocol 119
2.3.4.2 Layer 2 Tunneling Protocol 120
2.3.4.3 Secure Socket Tunneling Protocol 121
2.3.4.4 Microsoft Point-to-Point Encryption 121
2.3.4.5 OpenVPN 121
2.3.4.6 Internet Protocol Security 121
2.3.4.7 Secure Sockets Layer 122
2.3.5 Data loss prevention 122
2.3.6 Unified threat management 123
2.4 Endpoint security 126
2.4.1 Endpoint security basics 126
2.4.2 Malware protection 127
2.4.2.1 Signature-based anti-malware software 128
2.4.2.2 Container-based endpoint protection 130
2.4.2.3 Application whitelisting 130
2.4.2.4 Anomaly detection 131
2.4.3 Anti-spyware software 131
2.4.4 Personal firewalls 131
2.4.5 Host-based intrusion prevention systems 132
2.4.6 Mobile device management 132
2.5 Server and system administration 133
2.5.1 Identity and access management 134
2.5.2 Directory services 135
2.5.3 Vulnerability and patch management 135
2.5.4 Configuration management 136
2.5.5 Structured host and network troubleshooting 136
2.6 Secure the Enterprise (Strata) 140
2.6.1 Next-generation firewall 140
2.6.1.1 Application identification 143
App-ID traffic classification technology 145

App-ID: Addressing custom or unknown applications 146
App-ID in action: Identifying WebEx 148
Application identification and policy control 149
Application function control 149
Controlling multiple applications: Dynamic filters and groups 150
2.6.1.2 User identification 151
User-ID: Integrating user information and security policies 151
User-ID in action 152
Visibility into a user's activity 156
User-based policy control 157
Policy Optimizer 157
2.6.1.3 Content identification 157
Threat prevention 158
Stream-based malware scanning 159
Intrusion prevention 159
File and data filtering 160
2.6.1.4 Log correlation and reporting 160
2.6.1.5 Implementing Zero Trust with next-generation firewalls 168
Step 1: Define your protect surface 169
Step 2: Map the transaction flows 170
Step 3: Architect a Zero Trust network 170
Step 4: Create the Zero Trust policy 171
Step 5: Monitor and maintain the network 172
2.6.1.6 Next-generation firewall deployment options 174
Physical appliances (PA-Series) 174
Virtualized firewalls (VM-Series) 175
Cloud-native firewalls (CN-Series) 176
5G-ready firewalls (K2-Series) 178
2.6.1.7 IronSkillet 179
2.6.1.8 Palo Alto Networks Expedition (migration tool) 180
2.6.1.9 Best Practice Assessment (BPA) 181
2.6.2 Subscription services 182
2.6.2.1 DNS Security service 184

Predict and block new malicious domains 185
Neutralize DNS tunneling 186
Simplify security with automation and replace standalone tools 187
2.6.2.2 URL Filtering service 187
2.6.2.3 Threat Prevention service (antivirus, anti-spyware, and vulnerability protection) 189
Malware/antivirus protection 189
Command-and-control (spyware) protection 190
Vulnerability protection 190
2.6.2.4 Zero-day malware prevention (WildFire) 191
Behavior-based cyberthreat discovery 194
Threat prevention with global intelligence sharing 197
Integrated logging, reporting, and forensics 197
2.6.3 Network security management (Panorama) 198
2.6.4 Wi-Fi security (Okyo Garde) 204
Module 3 – Fundamentals of Cloud Security 206
3.0 Cloud Computing 206
3.0.1 Cloud Service Models 207
3.0.2 Cloud Deployment Models 207
3.0.3 Cloud Security Challenges 208
3.1 Cloud-Native Technologies 213
3.1.1 Virtualization 215
3.1.1.1 Virtual machines 216
3.1.1.2 Thin virtual machines 217
3.1.2 Containers and orchestration 217
3.1.2.1 VM-integrated containers 218
3.1.2.2 Containers 219
3.1.2.3 Containers as a Service 220
3.1.2.4 On-demand containers 221
3.1.3 Serverless computing 222
3.2 Cloud-Native Security 226
3.2.1 The 4C's of cloud native security 227

3.2.2 DevOps and DevSecOps 227
3.2.2.1 CI/CD pipeline 228
3.2.2.2 DevOps and security 229
3.2.3 Visibility, governance, and compliance 229
3.3 Hybrid Data Center Security 230
3.3.1 Traditional data security solution weaknesses 232
3.3.2 East-west traffic protection 233
3.3.3 Security in hybrid data centers 235
3.4 Secure the Cloud (Prisma) 238
3.4.1 Cloud application security (Prisma Cloud) 240
3.4.1.1 Cloud governance and compliance 241
3.4.1.2 Compute security 242
3.4.1.3 Network protection 243
3.4.1.4 Identity security 243
3.4.2 Secure Access Service Edge (Prisma Access) 244
3.4.2.1 Network-as-a-service layer 247
SD-WAN 248
Virtual private network 248
Zero Trust network access 249
Quality of service 250
3.4.2.2 Security-as-a-service layer 251
DNS security 251
Firewall as a service 251
Threat prevention 252
Secure web gateway 252
Data loss prevention 253
Cloud access security broker 254
3.4.3 Prisma SaaS 254
3.4.3.1 SaaS threat prevention 257
3.4.3.2 Data-exposure visibility 257
3.4.3.3 Contextual data-exposure control 257
3.4.3.4 Advanced document classification 258

3.4.3.5 Retroactive policy 258
3.5 Prisma Cloud Security Posture Management (CSPM) 259
Module 4 – Fundamentals of Security Operations 262
4.0 Elements of Security Operations 262
4.0.1 Business objectives 264
4.0.1.1 Mission 264
4.0.1.2 Governance 264
4.0.1.3 Planning 264
4.0.2 Business execution 265
4.0.2.1 Staffing 265
4.0.2.2 Career path progression 265
4.0.2.3 Employee utilization 266
4.0.2.4 Training 266
4.0.2.5 Facility 267
4.0.3 Business management and operations 267
4.0.3.1 Case management 267
4.0.3.2 Budget 268
4.0.3.3 Metrics 268
4.0.3.4 Reporting 269
4.0.3.5 Business liaisons 269
4.0.3.6 Governance, risk, and compliance 269
4.0.3.7 DevOps 269
4.1 Security Operations Processes 270
4.1.1 Identify 270
4.1.1.1 Alerting 271
4.1.1.2 Content engineering 271
4.1.1.3 Initial research 272
4.1.1.4 Severity triage 272
4.1.1.5 Escalation process 272
4.1.1.6 Alert distribution 273
4.1.2 Investigate 273
4.1.2.1 Detailed analysis 273

4.1.2.2 Forensics and telemetry 274
4.1.3 Mitigate 275
4.1.3.1 Mitigation 275
4.1.3.2 Preapproved mitigation scenarios 275
4.1.3.3 Breach response 275
4.1.3.4 Change control 276
4.1.3.5 Interface agreements 276
4.1.4 Improve 277
4.1.4.1 Tuning 277
4.1.4.2 Process improvement 277
4.1.4.3 Capability improvement 278
4.1.4.4 Quality review 278
4.2 Security Operations Infrastructure 279
4.2.1 Security information and event management 279
4.2.2 Analysis tools 280
4.2.3 SOC engineering 280
4.3 Security Operations Automation 280
4.3.1 Security orchestration, automation, and response 281
4.3.2 Security automation 282
4.4 Secure the Future (Cortex) 282
4.4.1 Endpoint protection (Cortex XDR) 284
4.4.1.1 Stop malware and ransomware 285
WildFire threat intelligence 286
Local analysis and machine learning 286
Behavioral threat protection 287
WildFire inspection and analysis 288
4.4.1.2 Block exploits and fileless threats 289
Pre-exploit protection 289
Technique-based exploit prevention 289
Kernel exploit prevention 290
Credential-theft protection 291
4.4.1.3 Investigate and respond to attacks 291

4.4.1.4 Extending prevention beyond Windows environments 292
Cortex XDR for macOS 292
Cortex XDR for Android 293
Cortex XDR for Linux 293
Device control for secure USB access 293
4.4.1.5 Simple endpoint security management 294
Cloud-based management 294
Intuitive interface 295
4.4.1.6 Benefits of a connected platform 295
Native integration for fast investigation and response 295
Coordinated enforcement 297
Centralized logging across the platform 297
4.4.1.7 Cortex XDR technical architecture 298
4.4.2 Cortex XSOAR 299
4.4.3 Threat intelligence (Cortex XSOAR TIM) 300
4.4.4 Cortex Data Lake 301
4.4.5 Cortex XSIAM 302
Appendix A – Knowledge Check Answers A-1
Section 1.0 Knowledge Check A-1
Section 1.1 Knowledge Check A-1
Section 1.2 Knowledge Check A-1
Section 1.3 Knowledge Check A-1
Section 1.4 Knowledge Check A-2
Section 2.0 Knowledge Check A-2
Section 2.1 Knowledge Check A-2
Section 2.2 Knowledge Check A-2
Section 2.3 Knowledge Check A-2
Section 2.4 Knowledge Check A-3
Section 2.5 Knowledge Check A-3
Section 2.6 Knowledge Check A-3
Section 3.0 Knowledge Check A-4

Section 3.1 Knowledge Check A-4
Section 3.2 Knowledge Check A-4
Section 3.3 Knowledge Check A-4
Section 3.4 Knowledge Check A-5
Section 4.0 Knowledge Check A-5
Section 4.1 Knowledge Check A-5
Section 4.2 Knowledge Check A-5
Section 4.3 Knowledge Check A-5
Section 4.4 Knowledge Check A-6

Appendix B – Glossary B-1

Appendix C – Palo Alto Networks Technical Training and Certification Programs C-1

Palo Alto Networks Technical Training Program C-1
Palo Alto Networks Certification Program C-1
Palo Alto Networks Certified Cybersecurity Entry-Level Technician (PCCET) C-1
Palo Alto Networks Certified Network Security Associate (PCNSA) C-1
Palo Alto Networks Certified Network Security Engineer (PCNSE) C-2

Table of Figures

Figure 1-1: The Cyberattack Lifecycle 28

Figure 1-2: Vulnerabilities can be exploited from the time software is deployed until it is patched. 42

Figure 1-3: Exploits rely on a series of core attack techniques to succeed. 43

Figure 1-4: The distributed C2 infrastructure of a botnet 46

Figure 1-5: Jasager pretends to be whichever access point is requested by the client's beacon. 57

Figure 1-6: Man-in-the-middle with SSLstrip 59

Figure 1-7: Zero Trust protect surface 66

Figure 1-8: Zero Trust conceptual architecture 69

Figure 1-9: Palo Alto Networks Security Operating Platform 73

Figure 2-1: DHCP operation 97

Figure 2-2: The OSI model and the TCP/IP model 111

Figure 2-3: Average time to detection by application vector 129

Figure 2-4: Palo Alto Networks next-generation firewalls use a single-pass architecture. 141

Figure 2-5: Next-generation firewall locations in the enterprise network 142

Figure 2-6: Application-centric traffic classification identifies specific applications on the network irrespective of the port and protocol in use. 144

Figure 2-7: How Palo Alto Networks App-ID classifies applications 146

Figure 2-8: Application function control maximizes productivity by safely enabling the application itself (Microsoft SharePoint) or individual functions. 150

Figure 2-9: User-ID integrates enterprise directories for user-based policies, reporting, and forensics. 152

Figure 2-10: Dynamic address groups (DAGs) 156

Figure 2-11: Stream-based scanning helps minimize latency and maximize throughput performance. 159

Figure 2-12: The ACC provides a highly visual, interactive, and customizable security management dashboard. 161

Figure 2-13: The ACC Application Usage widget displays application traffic by type, amount, risk, and category. 162

Figure 2-14: Geolocation awareness in the ACC provides valuable information about the source and destination of all application traffic. 162

Figure 2-15: The ACC Applications Using Non Standard Ports widget highlights port hopping and showcases the importance of application versus port control. 163

Figure 2-16: A wide variety of widgets can be selected to customize tabs in the ACC. 164

Figure 2-17: One-click, interactive capabilities provide additional information and the ability to apply any item as a global filter. 165

Figure 2-18: The automated correlation engine automatically highlights compromised hosts in the ACC by correlating indicators of compromise (IoCs). 166

Figure 2-19: The Palo Alto Networks Zero Trust methodology 169

Figure 2-20: Strata Next-Generation Firewalls 175

Figure 2-21: Due to the use of network address translation (NAT) in Kubernetes, all outbound traffic carries the node source IP address. 176

Figure 2-22: Security policies based on namespaces prevent spread of exploits within a physical cluster 177

Figure 2-23: The CN-Series deploys natively as control and dataplane pods within the Kubernetes environment. 178

Figure 2-24: Securing 4G and 5G New Radio (NR) networks 179

Figure 2-25: Security Capability Adoption Heatmap 182

Figure 2-26: Rich DNS data powers machine learning for protection. 185

Figure 2-27: URL Filtering service 188

Figure 2-28: Threat Prevention service 189

Figure 2-29: WildFire provides cloud-based malware analysis and threat prevention. 193

Figure 2-30: WildFire analysis 196

Figure 2-31: Panorama deployment modes 199

Figure 2-32: Panorama template stack and templates 200

Figure 2-33: Panorama device groups and policy evaluation 201

Figure 2-34: Integration with Splunk extends visibility and prevention capabilities to your entire network infrastructure 204

Figure 3-1: The shared responsibility model 208

Figure 3-2: The continuum of cloud-native technologies 214

Figure 3-3: VMs and thin VMs on the continuum of cloud-native technologies 217

Figure 3-4: VM-integrated containers on the continuum of cloud native technologies 219

Figure 3-5: Containers on the continuum of cloud-native technologies 220

Figure 3-6: CaaS platform on the continuum of cloud-native technologies 221

Figure 3-7: On-demand containers on the continuum of cloud-native technologies 222

Figure 3-8: Serverless architectures and the shared-responsibility model 223

Figure 3-9: Serverless on the continuum of cloud-native technologies 224

Figure 3-10: Data centers are evolving to include a mix of hardware and cloud computing technologies. 231

Figure 3-11: Typical virtual data center design architecture 233

Figure 3-12: Three-tier application hosted in a virtual data center 234

Figure 3-13: SASE delivers advanced network and security capabilities in a converged, cloud-delivered solution. 246

Figure 3-14: The Prisma Access architecture 247

Figure 3-15: Impacts of sanctioned and unsanctioned SaaS applications 255

Figure 3-16: Example of granular controls supported by App-ID 256

Figure 4-1: The purpose of security operations is to identify, investigate, and mitigate threats. 271

Figure 4-2: High-level view of how SOAR tools sit in a SOC 281

Figure 4-3: Struggles of a security analyst 283

Figure 4-4: Malicious files versus exploits 285

Figure 4-5: Cortex XDR leverages multiple technologies and techniques to protect endpoints from known and unknown malware. 286

Figure 4-6: Behavioral threat protection with Cortex XDR 287

Figure 4-7: Cortex XDR focuses on exploit techniques rather than on the exploits themselves. 290

Figure 4-8: Investigate and respond to attacks. 291

Figure 4-9: The Cortex XDR dashboard 294

Figure 4-10: Native integration with network, endpoint, and cloud apps as well as WildFire threat intelligence 296

Figure 4-11: Cortex XDR speeds alert triage and incident response. 297

Figure 4-12: Cortex XSOAR ingests alerts and IoCs from multiple detection sources and executes playbooks to enrich and respond to incidents. 300

Module 1 – Fundamentals of Cybersecurity

Knowledge Objectives

- Discuss modern computing trends, application threat vectors, cloud computing and software-as-a-service (SaaS) application challenges, data protection and privacy regulations and standards, and recent cyberattacks.
- Explain attacker motivations and the Cyberattack Lifecycle.
- Describe cyberattack techniques and types, including malware, vulnerabilities, exploits, spamming, phishing, bots and botnets, advanced persistent threats, and Wi-Fi attacks.
- Explain various network security models, concepts, and principles, including perimeter-based security and the Zero Trust model.
- Discuss the key capabilities of the Security Operating Platform and its key components.

1.0 Cybersecurity Landscape

The modern cybersecurity landscape is a rapidly evolving and hostile environment fraught with advanced threats and increasingly sophisticated threat actors. This section describes computing trends that are shaping the cybersecurity landscape, application frameworks and *attack* (or *threat*) *vectors*, cloud computing and SaaS application security challenges, various information security and data protection regulations and standards, and some recent cyberattack examples.

Note

The terms “enterprise” and “business” are used throughout this guide to describe organizations, networks, and applications in general. The use of these terms is not intended to exclude other types of organizations, networks, or applications, and should be understood to include not only large businesses and enterprises but also small and medium-size businesses (SMBs), government, state-owned enterprises (SOEs), public services, military, healthcare, and nonprofits, among others.

Key Terms

An *attack* (or *threat*) *vector* is a path or tool that an attacker uses to target a network.

1.0.1 Modern computing trends

The nature of enterprise computing has changed dramatically over the past decade. Core business applications are now commonly installed alongside *Web 2.0* apps on a variety of *endpoints*, and networks that were originally designed to share files and printers are now used to collect massive volumes of data, exchange real-time information, transact online business, and enable global collaboration.

Many Web 2.0 apps are available as *software-as-a-service* (SaaS), web-based, or as mobile apps that can be easily installed by end users or that can be run without installing any local programs or services on the endpoint. The use of Web 2.0 apps in the enterprise is sometimes referred to as *Enterprise 2.0*, although not all Web 2.0 apps are considered to be Enterprise 2.0 applications.

Key Terms

Web 2.0 is a term popularized by Tim O'Reilly and Dale Dougherty that unofficially refers to a new era of the World Wide Web, which is characterized by dynamic or user-generated content, interaction, and collaboration, as well as the growth of social media.

An *endpoint* is a computing device such as a desktop or laptop computer, handheld scanner, *internet of things* (IoT) device or sensor (such as an autonomous vehicle, smart appliance, smart meter, smart TV, or wearable device), point-of-sale (POS) terminal, printer, satellite radio, security or videoconferencing camera, self-service kiosk, smartphone, tablet, or *Voice over Internet Protocol* (VoIP) phone. Although endpoints can include servers and network equipment, the term is generally used to describe end-user devices.

The *internet of things* (IoT) is the network of physical smart objects that are embedded with electronics, software, sensors, and network connectivity to collect and share data.

Voice over IP (VoIP), or *IP telephony*, is technology that provides voice communication over an Internet Protocol (IP)-based network.

Software as a service (SaaS) is a category of cloud computing services in which the customer is provided access to a hosted application that is maintained by the service provider.

Enterprise 2.0 is a term introduced by Andrew McAfee and defined as "the use of emergent social software platforms within companies or between companies and their partners or customers."

Typical core business applications include:

- **Accounting software** is used to process and record accounting data and transactions such as accounts payable, accounts receivable, payroll, trial balances, and general ledger (GL) entries. Examples of accounting software include Intacct, Microsoft Dynamics AX and GP, NetSuite, QuickBooks, and Sage.

- **Business intelligence (BI) and business analytics software** consists of tools and techniques used to surface large amounts of raw unstructured data from a variety of sources (such as data warehouses and data marts). BI and business analytics software perform a variety of functions, including business performance management, data mining, event processing, and predictive analytics. Examples of BI and analytics software include IBM Cognos, MicroStrategy, Oracle Hyperion, and SAP.

- **Content management systems (CMS) and enterprise content management (ECM) systems** are used to store and organize files from a central management interface and include features such as indexing, publishing, search, workflow management, and versioning. Examples of CMS and ECM software include EMC Documentum, HP Autonomy, Microsoft SharePoint, and OpenText.

- **Customer relationship management (CRM)** software is used to manage an organization's customer (or client) information, including lead validation, past sales, communication and interaction logs, and service history. Examples of CRM suites include Microsoft Dynamics CRM, Salesforce.com, SugarCRM, and ZOHO.

- **Database management systems (DBMS)** are used to administer databases, including the schemas, tables, queries, reports, views, and other objects that comprise a database. Examples of DBMS software include Microsoft SQL Server, MySQL, NoSQL, and Oracle Database.

- **Enterprise resource planning (ERP)** systems provide an integrated view of core business processes such as product and cost planning, manufacturing or service delivery, inventory management, and shipping and payment. Examples of ERP software include NetSuite, Oracle's JD Edwards EnterpriseOne and PeopleSoft, and SAP.

- **Enterprise asset management (EAM)** software is used to manage an organization's physical assets throughout their entire lifecycle, including acquisition, upgrade, maintenance, repair, replacement, decommissioning, and disposal. EAM is commonly implemented as an integrated module of ERP systems. Examples of EAM software include IBM Maximo, Infor EAM, and SAP.

- **Supply chain management (SCM)** software is used to manage supply chain transactions, supplier relationships, and various business processes, such as purchase order processing, inventory management, and warehouse management. SCM software is commonly integrated with ERP systems. Examples of SCM software include Fishbowl Inventory, Freightview, Infor Supply Chain Management, and Sage X3.

- **Web content management (WCM)** software is used to manage website content, including administration, authoring, collaboration, and publishing. Examples of web content management software include Drupal, IBM FileNet, Joomla, and WordPress.

Common Web 2.0 apps and services (many of which are also SaaS apps) include:

- **File sync and sharing services** are used to manage, distribute, and provide access to online content, such as documents, images, music, software, and video. Examples include Apple iCloud, Box, Dropbox, Google Drive, Microsoft OneDrive, Spotify, and YouTube.

- **Instant messaging (IM)** is used to exchange short messages in real time. Examples include Facebook Messenger, Snapchat, and WhatsApp.

- **Microblogging** web services allow a subscriber to broadcast short messages to other subscribers. Examples include Tumblr and Twitter.

- **Office productivity suites** consist of cloud-based word processing, spreadsheet, and presentation software. Examples include Google Apps and Microsoft (Office) 365.

- **Remote access software** is used for remote sharing and control of an endpoint, typically for collaboration or troubleshooting. Examples include LogMeIn and TeamViewer.

- **Remote team meeting software** is used for audio conferencing, video conferencing, and screen sharing. Examples include Adobe Connect, Microsoft Teams, and Zoom.

- **Social curation** shares collaborative content about particular topics. Social bookmarking is also a type of social curation. Examples include Cogenz, Instagram, Pinterest, and Reddit.

- **Social networks** are used to share content with business or personal contacts. Examples include Facebook, Google+, and LinkedIn.

- **Web-based email** is an internet email service that is typically accessed via a web browser. Examples include Gmail, Outlook.com, and Yahoo! Mail.

- **Wikis** enable users to contribute, collaborate, and edit site content. Examples include Socialtext and Wikipedia.

Enterprise infrastructures (systems, applications, and networks) are rapidly converging with personal and Web 2.0 technologies and apps, making the definition of where the internet begins, and the enterprise infrastructure ends practically impossible. This convergence is being driven by several important trends, including:

- **5G cellular wireless.** Each new generation of wireless connectivity has driven a wealth of new innovations, and the move to the fifth-generation of cellular wireless (5G) is well underway, with mobile network operators announcing 5G pilot trials and commercialization plans as they expand their geographic footprints. The latest 5G applications are consumer-driven, help governments implement 5G for smart city rollouts, and bring 5G service experience to the public by seamlessly covering major sports events, among others. The promise of intelligent connectivity will drive a massive adoption of the internet of things (IoT) and has the potential to transform industries as well. We're now talking about the Enterprise of Things – networked industrial devices, sensors, networks, and apps that connect businesses. As today's enterprises undergo digital transformation, they'll be looking for 5G networks to drive true Industry 4.0 transformation, leveraging automation, *artificial intelligence* (AI), and IoT.

- **Bring your own apps (BYOA).** Web 2.0 apps on personal devices are increasingly being used for work-related purposes. As the boundary between work and personal lives becomes less distinct, end users are practically demanding that these same apps be available to them in their workplaces.

- **Bring your own device (BYOD).** Closely related to consumerization is BYOD, a policy trend in which organizations permit end users to use their own personal devices, primarily smartphones and tablets, for work-related purposes. BYOD relieves organizations from the cost of providing equipment to employees but creates a management challenge because of the vast number and type of devices that must be supported.

- **Cloud computing.** Cloud computing is now more ubiquitous than ever. According to the *Flexera 2021 State of the Cloud Report*, *public cloud* adoption is now at 97 percent for enterprises (1,000+ employees) and small-medium businesses (fewer than 1,000 employees), and *private cloud* adoption is at 80 percent. Additionally, 92 percent of enterprises have a *multicloud* strategy leveraging an average of more than five public

and/or private clouds.[1] Similarly, the Enterprise Strategy Group found that production server workloads increasingly run on a mix of cloud-ready architectures, including *virtual machines* (34 percent), *containers* (23 percent), and *serverless* (15 percent).[2]

- **Consumerization.** The process of consumerization occurs as end users increasingly find personal technology and apps that are more powerful or capable, more convenient, less expensive, quicker to install, and easier to use than enterprise IT solutions.

- **Content delivery networks (CDN).** Enterprises are using *content delivery networks* (CDNs) like Akamai, Amazon CloudFront, and Limelight networks to distribute their web products and services to customers worldwide. CDNs will grow even more prominent as 5G adoption continues to expand.

- **Managed security services.** The global shortage of cybersecurity professionals – estimated by the International Information System Security Certification Consortium (ISC)[2] to be 2.72 million in 2021 – is leading many organizations to partner with third-party security services organizations. These managed security service providers (MSSPs) typically operate a fully staffed 24/7 security operations centers (SOCs) and offer a variety of services such as log collection and aggregation in a security information and event management (SIEM) platform, event detection and alerting, vulnerability scanning and patch management, threat intelligence, and incident response and forensic investigation, among others.

- **Mobile computing.** The appetite for rapid, on-demand access to apps and data from anywhere, at any time, on any device is insatiable. There are now more than 8 billion mobile subscriptions worldwide, and total mobile monthly data traffic (including audio, file sharing, social networking, software uploads and downloads, video, web browsing, and other sources) is about 65 exabytes![3]

- **Work-from-home (WFH) and work-from-anywhere (WFA).** In the wake of the global pandemic, many organizations have implemented remote working models that include

---

[1] Flexera 2021 State of the Cloud Report." Accessed January 15, 2022. https://info.flexera.com/CM-REPORT-State-of-the-Cloud.

[2] Cahill, Doug. "Leveraging DevSecOps to Secure Cloud-native Applications." Enterprise Strategy Group. December 9, 2019. https://www.esg-global.com/research/esg-master-survey-results-leveraging-devsecops-to-secure-cloud-native-applications.

[3] "Ericsson Mobility Report, November 2021." Ericsson. Accessed January 16, 2022. https://www.ericsson.com/en/reports-and-papers/mobility-report/reports/november-2021.

WFH and WFA. In many cases, these organizations have realized additional benefits from these models, including increased operational efficiencies, higher employee productivity and morale, and greater access to a diverse talent pool that extends far beyond the immediate geographical region of the organization.

Key Terms

Artificial intelligence (AI) is the ability of a system or application to interact with and learn from its environment, and to automatically perform actions accordingly, without requiring explicit programming.

Public cloud is a cloud computing deployment model that consists of a cloud infrastructure that is open to use by the general public.

Private cloud is a cloud computing model that consists of a cloud infrastructure that is used exclusively by a single organization.

Multicloud is an enterprise cloud environment (or strategy) consisting of two or more public and/or private clouds.

A *virtual machine* (VM) is an emulation of a physical (hardware) computer system, including CPU, memory, disk, operating system, and network interfaces.

A *container* is a standardized, executable, and lightweight software code package that contains all the necessary components to run a given application (or applications) – including code, runtime, system tools and libraries, and configuration settings – in an isolated and virtualized environment to enable agility and portability of the application workloads.

Serverless generally refers to an operational model in cloud computing in which applications rely on managed services that abstract away the need to manage, patch, and secure infrastructure and virtual machines. Serverless applications rely on a combination of managed cloud services and function-as-a-service (FaaS) offerings.

A *content delivery network* (CDN) is a network of distributed servers that distributes cached webpages and other static content to a user from a geographic location that is physically closest to the user.

Moving beyond Web 2.0, *Web 3.0* will transform the enterprise computing landscape over the next decade and beyond. Web 3.0, as defined on ExpertSystem.com, is characterized by five main features:

- **Semantic web.** "The semantic web improves web technologies in order to generate, share and connect through search and analysis based on the ability to understand the meaning of words, rather than on keywords and numbers."
- **Artificial intelligence.** "Computers can understand information like humans in order to provide faster and more relevant results."
- **3D graphics.** 3D design is "used extensively in websites and services."
- **Connectivity.** "Information is more connected thanks to semantic metadata. As a result, the user experience evolves to another level of connectivity that leverages all the available information."
- **Ubiquity.** "Content is accessible by multiple applications, every device is connected to the web, [and] the services can be used everywhere."[4]

Key Terms

Web 3.0, as defined on ExpertSystem.com, is characterized by the following five characteristics: semantic web, artificial intelligence, 3D graphics, connectivity, and ubiquity.

For many, the vision of Web 3.0 is to return the power of the internet to individual users in much the same way that the original Web 1.0 was envisioned. To some extent, Web 2.0 has become shaped and characterized, if not controlled, by governments and large corporations dictating the content that is made available to individuals and raising many concerns about individual security, privacy, and liberty. Specific technologies that are evolving and beginning to form the foundations of Web 3.0 include (among others):

- AI and *machine learning* are two related technologies that enable systems to understand and act on information in much the same way that a human might use information. AI acquires and applies knowledge to find the most optimal solution, decision, or course of action. Machine learning is a subset of AI that applies algorithms to large datasets to discover common patterns in the data that can then be used to improve the performance of the system.
- *Blockchain* is essentially a data structure containing transactional records (stored as blocks) that ensures security and transparency through a vast, decentralized peer-to-

[4] Expert System. 2017. "5 main features of Web 3.0." Accessed January 16, 2022. http://www.expertsystem.com/web-3-0/.

peer network with no single controlling authority. *Cryptocurrency*, such as Bitcoin, is an example of a blockchain application.

- *Data mining* enables patterns to be discovered in large datasets by using machine learning, statistical analysis, and database technologies.
- *Mixed reality* includes technologies, such as *virtual reality* (VR), *augmented reality* (AR), and *extended reality* (XR), that deliver an immersive and interactive physical and digital sensory experience in real time.
- *Natural language search* is the ability to understand human spoken language and context, rather than a *Boolean* search, for example, to find information.

Key Terms

Machine learning is a subset of AI that applies algorithms to large datasets to discover common patterns in the data that can then be used to improve the performance of the system.

Blockchain is essentially a data structure containing transactional records (stored as blocks) that ensures security and transparency through a vast, decentralized peer-to-peer network with no single controlling authority. Cryptocurrency is an internet-based financial instrument that uses blockchain technology.

Data mining enables patterns to be discovered in large datasets by using machine learning, statistical analysis, and database technologies.

Mixed reality (MR) includes technologies, such as virtual reality (VR), augmented reality (AR), and extended reality (XR), that deliver an immersive and interactive physical and digital sensory experience in real time. *Virtual reality* is a simulated experience. *Augmented reality* enhances a real-world environment with virtual objects. *Extended reality* broadly covers the spectrum from physical to virtual reality with various degrees of partial sensory to fully immersive experiences.

Natural language search is the ability to understand human spoken language and context, rather than a Boolean search, for example, to find information. *Boolean* refers to a system of algebraic notation used to represent logical propositions.

Organizations are often unsure of the potential business benefits – and the inherent risks – of new trends such as Web 2.0 and Web 3.0, and therefore either:

- Implicitly allow personal technologies and apps by simply ignoring their use in the workplace, or
- Explicitly prohibit their use but are then unable to effectively enforce such policies with traditional firewalls and security technologies

Whether personal technologies and apps are implicitly allowed (and ignored) or explicitly prohibited (but not enforced), the adverse results of ineffective policies can include:

- **Lost productivity** because users must either find ways to integrate these unsupported technologies and apps (when allowed) with the enterprise infrastructure or use applications that are unfamiliar to them or less efficient (when personal technologies and apps are prohibited)
- **Potential disruption of critical business operations** because of underground or back-channel processes that are used to accomplish specific workflow tasks or to circumvent controls, and are known to only a few users and are fully dependent on their use of personal technologies and apps
- **Exposure to additional risks** for the enterprise due to unknown – and therefore unpatched – vulnerabilities in personal technologies and apps, and a perpetual cat-and-mouse game between employees who circumvent controls (for example, with external proxies, encrypted tunnels, and remote desktop applications) and security teams that manage these risks
- **Penalties for regulatory non-compliance**, for example, the EU General Data Protection Regulation (GDPR), the U.S. Health Insurance Portability and Accountability Act (HIPAA), and the Payment Card Industry Data Security Standard (PCI DSS)

As these trends continue to blur the distinction between the internet and the enterprise network, new security challenges and risks emerge, including:

- New application threat vectors
- Turbulence in the cloud
- SaaS application risks

1.0.2 New application threat vectors

Exploiting vulnerabilities in core business applications has long been a predominant attack vector, but threat actors are constantly developing new tactics, techniques, and procedures (TTPs). To effectively protect their networks and cloud environments, enterprise security teams

must not only manage the risks associated with a relatively limited, known set of core applications but also manage the risks associated with an ever-increasing number of known and unknown cloud-based applications. The cloud-based application consumption model has revolutionized the way organizations do business, and applications such as Microsoft (Office) 365 and Salesforce are being consumed and updated entirely in the cloud.

Classifying applications as either “good” (allowed) or “bad” (blocked) in a clear and consistent manner has also become increasingly difficult. Many applications are clearly good (low risk, high reward) or clearly bad (high risk, low reward), but most are somewhere in between depending on how the application is being used.

For example, many organizations use social networking applications such as Facebook, LinkedIn, and Twitter for important business functions such as recruiting, research and development, marketing, and consumer advocacy. However, these same applications can be used to leak sensitive information or cause damage to an organization’s public image, whether inadvertently or maliciously.

Many applications are designed to circumvent traditional port-based firewalls (discussed in Section 2.3.1), so that they can be easily installed and accessed on any device, anywhere and anytime, using techniques such as:

- **Port hopping**, in which ports and protocols are randomly changed during a session.
- **Use of non-standard ports**, such as running Yahoo! Messenger over TCP port 80 (HTTP) instead of the standard TCP port for Yahoo! Messenger (5050).
- **Tunneling within commonly used services**, such as when peer-to-peer (P2P) file sharing or an instant messenger (IM) client such as Meebo is running over HTTP.
- **Hiding within SSL encryption**, which masks the application traffic, for example, over TCP port 443 (HTTPS). More than half of all web traffic is now encrypted.

Many traditional client-server business applications are also being redesigned for web use and employ these same techniques for ease of operation while minimizing disruptions. For example, both *remote procedure call* (RPC) and Microsoft SharePoint use port hopping because it is critical to how the protocol or application (respectively) functions, rather than as a means to evade detection or enhance accessibility.

Applications can also be hijacked and repurposed by malicious actors, such as was done in the 2014 Heartbleed attack. According to an April 2014 Palo Alto Networks article:

Key Terms

Remote procedure call (RPC) is an inter-process communication (IPC) protocol that enables an application to be run on a different computer or network, rather than the local computer on which it is installed.

"[T]he story of Heartbleed's impact has been focused on the compromise of HTTPS-enabled websites and web applications, such as Yahoo!, Google, Dropbox, Facebook, online banking, and the thousands of other vulnerable targets on the web. These are of huge impact, but those sites will all be updated quickly….

"For security professionals, [the initial Heartbleed attack] is only the tip of the iceberg. The vulnerability puts the tools once reserved for truly advanced threats into the hands of the average attacker – notably, the ability to breach organizations, and move laterally within them. Most enterprises of even moderate size do not have a good handle on what services they are running internally using SSL encryption. Without this baseline knowledge, it is extremely difficult for security teams to harden their internal attack surface against the credential and data stealing tools Heartbleed enables. All footholds for the attacker with an enterprise network are suddenly of equal value."[5]

As new applications are increasingly web-enabled and browser-based, HTTP and HTTPS now account for about two-thirds of all enterprise network traffic. Traditional port-based firewalls and other security infrastructure cannot distinguish whether these applications, riding on HTTP and HTTPS, are being used for legitimate business purposes.

Thus, applications (including malware) have become the predominant attack vector to infiltrate networks and systems.

1.0.3 Turbulence in the cloud

Cloud computing technologies enable organizations to evolve their data centers from a hardware-centric architecture where applications run on dedicated servers to a dynamic and

[5] Simkin, Scott. "Real-world Impact of Heartbleed (CVE-2014-0160): The Web is Just the Start." Palo Alto Networks. April 10, 2014. https://researchcenter.paloaltonetworks.com/2014/04/real-world-impact-heartbleed-cve-2014-0160-web-just-start.

automated environment where pools of computing resources are available on demand, to support application workloads that can be accessed anywhere, anytime, and from any device.

However, many organizations have been forced into significant compromises regarding their public and private cloud environments – trading function, visibility, and security for simplicity, efficiency, and agility. If an application hosted in the cloud isn't available or responsive, network security controls, which all too often introduce delays and outages, are typically "streamlined" out of the cloud design. Cloud security trade-offs often include

- Simplicity *or* function
- Efficiency *or* visibility
- Agility *or* security

Many of the features that make cloud computing attractive to organizations also run contrary to network security best practices. For example:

- **Cloud computing doesn't mitigate existing network security risks.** The security risks that threaten your network today don't go away when you move to the cloud. The shared responsibility model defines who (customer and/or provider) is responsible for what (related to security) in the public cloud. In general terms, the cloud provider is responsible for security *of* the cloud, including the physical security of the cloud data centers, and for foundational networking, storage, compute, and virtualization services. The cloud customer is responsible for security *in* the cloud, which is further delineated by the cloud service model. For example, in an infrastructure-as-a-service (IaaS) model, the cloud customer is responsible for the security of the operating systems, middleware, runtime, applications, and data. In a platform-as-a-service (PaaS) model, the cloud customer is responsible for the security of the applications and data – the cloud provider is responsible for the security of the operating systems, middleware, and runtime. In a SaaS model, the cloud customer is responsible only for the security of the data, and the cloud provider is responsible for the full stack, from the physical security of the cloud data centers to the application.

- **Separation and segmentation are fundamental to security; the cloud relies on shared resources.** Security best practices dictate that mission-critical applications and data be separated in secure segments on the network, based on Zero Trust principles (discussed in Section 1.3.2). On a physical network, Zero Trust is relatively straightforward, using firewalls and policies based on application and user identity. In a cloud environment, direct communication between virtual machines (VMs) within a server host occurs constantly – in some cases, across varied levels of trust, thus making segmentation a

real challenge. Mixed levels of trust, combined with a lack of intra-host traffic visibility by virtualized port-based security offerings, may weaken your security posture.

- **Security deployments are process-oriented; cloud computing environments are dynamic.** The creation or modification of your cloud workloads can often be done in minutes, yet the security configuration for this workload may take hours, days, or weeks. Security delays aren't designed to be burdensome; they're the result of a process that is designed to maintain a strong security posture. Policy changes need to be approved, the appropriate firewalls need to be identified, and the relevant policy updates need to be determined. In contrast, the cloud is a highly dynamic environment, with workloads being added, removed, and changed rapidly and constantly. The result is a disconnect between security policy and cloud workload deployments, which leads to a weakened security posture. Thus, security technologies and processes must be able to auto scale to take advantage of the elasticity of the cloud while maintaining a strong security posture.

- **Infrastructure as code automates the ability to rapidly scale secure configurations and misconfigurations.** Organizations are rapidly adopting *infrastructure as code* (IaC) as they attempt to automate more of their build processes in the cloud. IaC has become popular as it enables immutable infrastructure. This is the ability to standardize and freeze many parts of cloud infrastructure, so results are consistent and predictable when running code every time. For example, if you know that every node in your cloud has the exact same virtual networking configuration, your chances of having networking-related app problems decreases significantly. And while IaC offers security teams a predictable way to enforce security standards, this powerful capability remains largely unharnessed. The challenge for organizations is ensuring that IaC configurations are consistently enforced across multiple public cloud accounts, providers, and software development pipelines.

- **Data can be quickly and easily consumed by applications and users in the cloud.** However, more sophisticated threats and new privacy regulations have raised the stakes on data security everywhere – including in the cloud. Data loss prevention (DLP) provides visibility across all sensitive information, everywhere and at all times, enabling strong protective actions to safeguard data from threats and violations of corporate policies. But legacy standalone DLP technologies are not efficient for today's cloud-driven world. Built on old core engines specifically for on-premises environments, the technology has not changed significantly in the last decade. To adjust to cloud initiatives, legacy DLP providers are simply extending their existing solutions to cloud environments, which creates a gap in visibility and management and minimizes policy

control. Organizations that have spent enormous amounts of time and money to build a custom DLP architecture to fit their network environments are now struggling with complexity and poor usability as they try to "add in" their cloud apps, data, and public cloud instances. Additionally, security teams face the challenge of using effective but complex DLP technologies while balancing the constant work that comes with them, from ongoing policy tuning to exhausting incident triage cycles and incident response decisions. These teams are drowning in too many alerts – most of which turn out to be false positives – and often respond to a data incident too late.

1.0.4 SaaS application risks

Data is located everywhere in today's enterprise networks, including in many locations that are not under the organization's control. New data security challenges emerge for organizations that permit SaaS use in their networks.

Key Terms

Infrastructure as code (IaC) is a *DevOps* process in which developers or IT operations teams can programmatically provision and manage the infrastructure stack (such as virtual machines, networks, and connectivity) for an application in software.

DevOps is the culture and practice of improved collaboration between application development and IT operations teams.

With SaaS applications, data is often stored where the application resides – in the cloud. Thus, the data is no longer under the organization's control, and visibility is often lost. SaaS vendors do their best to protect the data in their applications, but it is ultimately not their responsibility. Just as in any other part of the network, the IT team is responsible for protecting and controlling the data, regardless of its location.

Because of the nature of SaaS applications, their use is very difficult to control – or have visibility into – after the data leaves the network perimeter. This lack of control presents a significant security challenge: End users are now acting as their own "shadow" IT department, with control over the SaaS applications they use and how they use them. But they have little or no understanding of the inherent data exposure and threat insertion risks of SaaS, including:

- **Malicious outsiders.** The most common source of breaches for networks overall is also a critical concern for SaaS security. The SaaS application becomes a new threat vector and distribution point for malware used by external adversaries. Some malware will even

target the SaaS applications themselves, for example, by changing their shares to "public" so that the data can be retrieved by anyone.

- **Accidental data exposure.** Well-intentioned end users are often untrained and unaware of the risks their actions pose in SaaS environments. Because SaaS applications are designed to facilitate easy sharing, it's understandable that data often becomes unintentionally exposed. Accidental data exposure by end users is surprisingly common and includes:

    - **Accidental share.** A share meant for a particular person is accidentally sent to the wrong person or group. Accidental shares are common when a name auto fills, or is mistyped, which may cause an old email address or the wrong name, group, or even an external user, to have access to the share.

    - **Promiscuous share.** A legitimate share is created for a user, but that user then shares with other people who shouldn't have access. Promiscuous shares often result in the data being publicly shared because it can go well beyond the control of the original owner.

    - **Ghost (or stale) share.** A share remains active for an employee or vendor that is no longer working with the company, or should no longer have access. Without visibility and control of the shares, the tracking and fixing of shares to ensure that they are still valid is very difficult.

- **Malicious insiders.** The least common but real SaaS application risk is the internal user who maliciously shares data for theft or revenge purposes. For example, an employee who is leaving the company might set a folder's share permissions to "public" or share it with an external email address to later steal the data from a remote location.

The average enterprise has 288 SaaS apps in use.[6] It is important to consider the security of the apps, what data they have access to, and how employees are using them. Here are several best practices for securing sensitive data in SaaS apps:

- **Discover employee use of unvetted SaaS applications.** As SaaS adoption rapidly expands, manual discovery of SaaS usage in the enterprise becomes increasingly untenable. Instead, to quickly identify risk – and extend appropriate security controls – your organization needs an automated way to continuously discover all SaaS applications in use by employees.

[6] "SaaS Trends 2020." Blissfully. October 23, 2019. https://blissfully.com/saas-trends/2020-annual-report/.

- **Protect sensitive data in SaaS applications.** Implement advanced DLP capabilities using an *application programming interface* (API)-based approach to scan for sensitive information stored within SaaS applications. Compared to inline, an API-based approach provides deeper context and allows for automatic remediation of data-risk violations.
- **Secure your weakest link – SaaS users.** Start with user training and interactive coaching to identify and help change risky behavior. Then, give your security team tools to help them monitor and govern SaaS application permissions. Look for a solution with robust access controls, including:
    - *Multi-factor authentication* (MFA)
    - *Role-based access control* (RBAC)
    - Protection for administrative accounts
    - User access monitoring that can detect malicious or risky behavior
- **Enforce compliance requirements in the cloud.** Create and enforce a consistent, granular security policy for compliance that covers all SaaS applications used by your organization. This includes automating compliance and reporting for all relevant regulatory requirements across your SaaS applications.
- **Reduce risk from unmanaged devices.** Deploy a security product that differentiates access between managed and unmanaged devices to protect against the increased security risks inherent with personal devices. For instance, you could allow downloads to managed devices but block them for unmanaged ones while enabling access to core functionality.
- **Control data sharing from SaaS applications.** Use an inline approach to gain visibility into sensitive data flowing into high-risk, unsanctioned applications. Create and enforce DLP policies that control data-sharing activities in the SaaS applications employees use.
- **Stop SaaS-borne malware threats.** Implement threat prevention technology that works with your SaaS security to block malware and stop threats from spreading through SaaS applications, eliminating a new insertion point for malware.

Key Terms

An *application programming interface* (API) is a set of routines, protocols, and tools for building software applications and integrations.

Multi-factor authentication (MFA) refers to any authentication mechanism that requires two or more of the following factors: something you know, something you have, and/or something you are.

Role-based access control (RBAC) is a method for implementing discretionary access controls in which access decisions are based on group membership, according to organizational or functional roles.

1.0.5 Compliance and security are not the same

A rapidly and ever-increasing number of international, multinational, federal, regional, state, and local laws and regulations mandate numerous cybersecurity and data protection requirements for businesses and organizations worldwide. Various industry directives, such as the Payment Card Industry Data Security Standard (PCI DSS), also establish their own cybersecurity standards and best practices for businesses and organizations operating under their purview.

This complex regulatory environment is further complicated by the fact that many laws and regulations are obsolete, ambiguous, not uniformly supported by international communities, and/or inconsistent (with other applicable laws and regulations), thus requiring legal interpretation to determine relevance, intent, and/or precedence. As a result, businesses and organizations in every industry struggle to achieve and maintain compliance.

You should understand that compliance and security are not the same thing. An organization can be fully compliant with the various cybersecurity laws and regulations that are applicable for that organization, yet still not be secure. Conversely, an organization can be secure yet not be fully compliant. As if to underscore this point, the compliance and security functions in many organizations are separate.

Pertinent examples (neither comprehensive nor exhaustive) of current cybersecurity laws and regulations include:

- **Australian Privacy Principles.** The Privacy Act 1988 establishes standards for collecting and handling personal information, referred to as the Australian Privacy Principles (APP).

- **California Consumer Privacy Act (CCPA).** A privacy rights and consumer protection statute for residents of California that was enacted in 2018 and became effective on January 1, 2020.

- **California Privacy Rights Act (CPRA).** Sometimes referred to as "CCPA 2.0", the CPRA took effect in December 2020 and becomes operative in January 2023. CPRA significantly amends and expands CCPA.

- **Canada Personal Information Protection and Electronic Documents Act (PIPEDA).** PIPEDA defines individual rights with respect to the privacy of their personal information and governs how private sector organizations collect, use, and disclose personal information in the course of business.

- **Colorado Privacy Act (CPA).** The CPA takes effect in July 2023 and protects the personal data of Colorado residents.

- **European Union (EU) General Data Protection Regulation (GDPR).** The GDPR applies to any organization that does business with EU residents. It strengthens data protection for EU residents and addresses the export of personal data outside the EU.

- **EU Network and Information Security (NIS) Directive:** An EU directive that imposes network and information security requirements for banks, energy companies, healthcare providers, and digital service providers, among others.

- **North American Electric Reliability Corporation (NERC) Critical Infrastructure Protection (CIP).** NERC CIP defines cybersecurity standards to protect the physical and cyber assets necessary to operate the bulk electric system (BES) – the power grid – in the United States and Canada. The standards are mandatory for all BES-generating facilities with different criteria based on a tiered classification system (high, medium, or low impact).

- **Payment Card Industry Data Security Standard (PCI DSS).** PCI DSS applies to any organization that transmits, processes, or stores payment card (such as debit and credit cards) information. PCI DSS is mandated and administered by the PCI Security Standards Council (SSC) comprising Visa, MasterCard, American Express, Discover, and JCB.

- **U.S. Cybersecurity Enhancement Act of 2014.** This act provides an ongoing, voluntary public-private partnership to improve cybersecurity and to strengthen cybersecurity research and development, workforce development and education, and public awareness and preparedness.

- **U.S. Cybersecurity Information Sharing Act (CISA).** This act enhances information sharing about cybersecurity threats by allowing internet traffic information to be shared between the U.S. government and technology and manufacturing companies.

- **U.S. Federal Exchange Data Breach Notification Act of 2015.** This act further strengthens HIPAA by requiring health insurance exchanges to notify individuals whose personal information has been compromised as the result of a data breach as soon as possible but no later than 60 days after breach discovery.

- **U.S. Federal Information Security Modernization Act (FISMA).** Known as the Federal Information Security Management Act prior to 2014, FISMA implements a comprehensive framework to protect information systems used in federal government agencies.

- **U.S. Gramm-Leach-Bliley Act (GLBA).** Also known as the Financial Services Modernization Act of 1999, relevant provisions of GLBA include the Financial Privacy Rule and the Safeguards Rule, which require financial institutions to implement privacy and information security policies to safeguard the non-public personal information of clients and consumers.

- **U.S. Health Insurance Portability and Accountability Act (HIPAA).** The HIPAA Privacy Rule establishes national standards to protect individuals' medical records and other personal health information. It requires appropriate safeguards for *protected health information* (PHI) and applies to *covered entities* and their business associates.

- **U.S. National Cybersecurity Protection Advancement Act of 2015.** This act amends the Homeland Security Act of 2002 to enhance multidirectional sharing of information related to cybersecurity risks and strengthens privacy and civil liberties protections.

- **U.S. Sarbanes-Oxley (SOX) Act.** This act was enacted to restore public confidence following several high-profile corporate accounting scandals, most notably Enron and Worldcom. SOX increases financial governance and accountability in publicly traded companies. Section 404 of SOX specifically addresses internal controls, including requirements to safeguard the confidentiality, integrity, and availability of IT systems.

- **Virginia Consumer Data Protection Act (VCDPA).** The VCDPA takes effect in January 2023 and protects the personal data of Virginia residents.

Key Terms

Protected health information (PHI) is defined by HIPAA as information about an individual's health status, provision of healthcare, or payment for healthcare that includes identifiers such as names, geographic identifiers (smaller than a state), dates, phone and fax numbers, email addresses, Social Security numbers, medical record numbers, and photographs.

A *covered entity* is defined by HIPAA as a healthcare provider that electronically transmits PHI (such as doctors, clinics, psychologists, dentists, chiropractors, nursing homes, and pharmacies), a health plan (such as a health insurance company, health maintenance organization, company health plan, or government program, including Medicare, Medicaid, and military and veterans' healthcare), or a healthcare clearinghouse.

1.0.6 Recent high-profile cyberattack examples

Thousands of cyberattacks are perpetrated against enterprise networks every day. Unfortunately, many more of these attacks succeed than are typically reported in the mass media. For organizations that are the victims of such attacks, the financial and reputational damage can be devastating. Some high-profile past breaches that continue to serve as cautionary examples many years later include:

- **Target.** In late 2013, Target discovered that credit card data and debit card data from 40 million of its customers, and the personal information of an additional 70 million of its customers, had been stolen over a period of about 19 days, from November 27 to December 15, 2013. The attackers were able to infiltrate Target's point-of-sale (POS) systems by installing malware (believed to be a variant of the ZeuS financial botnet) on an HVAC (heating, ventilation, and air conditioning) contractor's computer systems to harvest credentials for an online portal used by Target's vendors. Target's 2016 annual report disclosed that the total cost of the breach was US$292 million.

- **Home Depot.** In September 2014, Home Depot suffered a data breach that went unnoticed for about five months. As with the Target data breach, the attackers used a vendor's credentials and exploited a *zero-day threat*, based on a Windows vulnerability, to gain access to Home Depot's network. Memory scraping malware was then installed on more than 7,500 self-service POS terminals to collect 56 million customer credit card numbers throughout the United States and Canada. Home Depot's 2016 annual report disclosed that the total cost of the breach was US$298 million.

- **Anthem.** In February 2015, Anthem disclosed that its servers had been breached and *personally identifiable information* (PII) including names, Social Security numbers, birthdates, addresses, and income information for about 80 million customers had been stolen. The breach occurred on December 10, 2014, when attackers compromised an Anthem database by using a database administrator's credentials. The breach wasn't found until January 27, 2015, when the database administrator discovered a questionable query being run with his credentials. The total cost of the breach is expected to reach US$31 billion.

Key Terms

A *zero-day threat* is the window of vulnerability that exists from the time a new (unknown) threat is released until security vendors release a signature file or security patch for the threat.

Personally identifiable information (PII) is defined by the U.S. National Institute of Standards and Technology (NIST) as "any information about an individual maintained by an agency, including (1) any information that can be used to distinguish or trace an individual's identity … and (2) any other information that is linked or linkable to an individual…." Examples of PII include:

- **Name** (such as full name, maiden name, mother's maiden name, or alias)
- **Personal identification number** (such as Social Security number, passport number, driver's license number, financial account number, or credit card number)
- **Address information** (such as street address or email address)
- **Telephone numbers** (such as mobile, business, and personal numbers)
- **Personal characteristics** (such as photographs, X-rays, fingerprints, and biometric data)
- **Information about personally owned property** (such as vehicle registration number and title information)
- **Information that is linked or linkable to any of the above** (such as birthdate, birthplace, religion, and employment, medical, education, and financial records)

Several other recent examples of attacks and breaches include:

- **Marriott.** In November 2018, Marriott reported a data breach potentially involving the credit card information, passport numbers, and other personal data of up to 500 million

hotel guests of more than 6,700 properties in its Starwood hotel brands (W Hotels, St. Regis, Sheraton, Westin, Element, Aloft, The Luxury Collection, Le Méridien, and Four Points) over a four-year period from 2014 to 2018. The sensitive nature of the personal data – which included mailing addresses, phone numbers, email addresses, dates of birth, gender, reservation dates, and arrival and departure dates/times – opened the door to a broad range of potential criminal activities beyond credit card fraud and identity theft.

- **Quest Diagnostics.** In May 2019, Quest Diagnostics was notified by one of its billing collections service providers, American Medical Collection Agency (AMCA), that an unauthorized user had potentially accessed more than 12 million patient records including individual patient records, financial data, Social Security numbers, and other medical information.

- **City of Baltimore.** The U.S. city of Baltimore, Maryland, was hit by a ransomware attack in May 2019, demanding payment of $72,000 in bitcoin. Although the city appropriately refused to pay the ransom, they have budgeted $18.2 million to remediate the damage associated with the attack. Baltimore is just one example: 82 U.S. cities and municipalities were hit by ransomware attacks in 2019.

- **Capital One.** In July 2019, Capital One announced a data breach affecting more than 100 million individual customers in the U.S. and Canada, which resulted from an individual exploiting a configuration vulnerability. Although the breach did not compromise credit card numbers or account login credentials, it exposed PII and other sensitive information including names, addresses, phone numbers, email addresses, dates of birth, some Social Security numbers, self-reported incomes, credit scores, credit limits and balances, payment history, and transaction data.

- **Gekko Group.** In November 2019, France-based Gekko Group, a subsidiary of Accor Hotels, suffered a data breach in a database containing over 1 terabyte of data. The breach potentially exposed the customer information of Gekko Group brands (600,000 hotels worldwide), their clients, and connected external websites and platforms (such as Booking.com), including PII, hotel and transport reservations, and credit card information.

- **SolarWinds.** In December 2020, the cybersecurity firm FireEye and the U.S. Treasury Department both reported attacks involving malware in a software update to their SolarWinds Orion Network Management System perpetrated by the APT29 (Cozy Bear/Russian SVR) threat group. This attack is one of the most damaging supply chain

attacks in history, potentially impacting more than 300,000 SolarWinds customers, including the U.S. federal government and 425 of the Fortune 500 companies.

- **Colonial Pipeline.** In May 2021, the Colonial Pipeline Company – which operates one of the largest fuel pipelines in the U.S. – was hit by the DarkSide threat actor group with a ransomware-as-a-service (RaaS) attack. Although the company acted quickly to shut down its network systems and paid the $4.4 million ransom, operations were not fully restored for six days, which caused major fuel shortages and other supply chain issues along the U.S. eastern seaboard. Additionally, the personal information – including the health insurance, Social Security, driver's license, and military identification numbers – of nearly 6,000 individuals was compromised.

- **JBS S.A.** In May 2021, Brazil-based JBS S.A. – the largest producer of beef, chicken, and pork worldwide – was hit by a ransomware attack attributed to the REvil threat actor group. Although the company paid the $11 million ransom, its U.S. and Australia beef processing operations were shut down for a week.

- **Government of Ukraine.** In January 2022, several Ukrainian government websites including the ministry of foreign affairs and the education ministry were hacked by suspected Russian attackers. Threatening messages were left on the websites during a period of heightened tensions between the governments of Ukraine and Russia.

Important lessons to be learned from these attacks include:

- A "low and slow" cyberattack can go undetected for weeks, months, or even years.

- An attacker doesn't necessarily need to run a sophisticated exploit against a hardened system to infiltrate a target organization. Often, an attacker will target an auxiliary system or other vulnerable endpoint, then pivot the attack toward the primary target.

- Unpatched vulnerabilities are a commonly exploited attack vector.

- The direct and indirect financial costs of a breach can be devastating for both the targeted organization and individuals whose personal and financial information is stolen or compromised.

1.0 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **True or False.** Business intelligence (BI) software consists of tools and techniques used to surface large amounts of raw unstructured data to perform a variety of tasks, including data mining, event processing, and predictive analytics.

2. **True or False.** The process in which end users find personal technology and apps that are more powerful or capable, more convenient, less expensive, quicker to install, and easier to use than enterprise IT solutions is known as *consumerization*.

3. **True or False.** An organization can be compliant with all applicable security and privacy regulations for its industry yet still not be secure.

4. **Fill in the Blank.** The U.S. law that establishes national standards to protect individuals' medical records and other health information is known as the _____ _____.

5. **Classroom Discussion.** What are lessons or common themes that can be derived from the Target, Anthem, SolarWinds, and Colonial Pipeline cyberattack examples?

1.1 Cyberthreats

This section describes cybersecurity adversaries – the various threat actors, their motivations, and the cyberattack strategy.

1.1.1 Attacker profiles and motivations

In *The Art of War*, Sun Tzu teaches "know thy enemy, know thy self. A thousand battles, a thousand victories" (translated in various forms) to instill the importance of understanding the strengths, weaknesses, strategies, and tactics of your adversary as well as you know your own. Of course, in modern cyber warfare a thousand battles can occur in a matter of seconds, and a single victory by your enemy can imperil your entire organization. Thus, knowing your enemies – including their means and motivations – is more important than ever.

In the relatively innocuous "good ol' days" of *hackers* and *script kiddies*, the primary motivation for a cyberattack was notoriety, and the attack objective was typically limited to defacing or "owning" a website to cause inconvenience and/or embarrassment to the victim.

Key Terms

The term *hacker* was originally used to refer to anyone with highly specialized computing skills, without connoting good or bad purposes. However, common misuse of the term has redefined a hacker as someone who circumvents computer security with malicious intent, such as a cybercriminal, cyberterrorist, or hacktivist, cracker, and/or black hat.

A *script kiddie* is someone with limited hacking and/or programming skills who uses malicious programs (malware) written by others to attack a computer or network.

Modern cyberattacks are perpetrated by far more sophisticated and dangerous adversaries, motivated by far more sinister purposes:

- **Cybercriminals.** Acting independently or as part of a criminal organization, cybercriminals commit acts of data theft, embezzlement, fraud, and/or extortion for financial gain. According to the RAND Corporation, "In certain respects, the black market [for cybercrime] can be more profitable than the illegal drug trade,"[7] and by many estimates, cybercrime is now a US$1 trillion industry.

- **State-affiliated groups.** Sponsored by or affiliated with nation-states, these organizations have the resources to launch very sophisticated and persistent attacks, have great technical depth and focus, and are well funded. They often have military and/or strategic objectives such as the ability to disable or destroy critical infrastructure, including power grids, water supplies, transportation systems, emergency response, and medical and industrial systems. The Center for Strategic and International Studies reports that "At the nation-state level, Russia, Iran, and North Korea are using coercive cyberattacks to increase their sphere of influence, while China, Russia, and Iran have

[7] Lillian Ablon, Martin Libicki, and Andrea Golay. "Markets for Cybercrime Tools and Stolen Data." RAND Corporation, National Security Research Division. Accessed January 16, 2022. https://www.rand.org/content/dam/rand/pubs/research_reports/RR600/RR610/RAND_RR610.pdf.

conducted reconnaissance of networks critical to the operation of the U.S. power grid and other critical infrastructure without penalty."[8]

- **Cybercrime vendors.** Capitalizing on the service model of cloud computing, many threat actors now rent or sell their malware and exploits – including *business email compromise (BEC)* and ransomware – as cybercrime-as-a-service (CCaaS) offerings on the dark web. Vendors profit from the purchase or rental of their services and potentially earn a commission from the attacks themselves. Additional services often include mix-and-match bundles, collection services, volume discounts, and 24-hour support.

- **Hacktivists.** Motivated by political or social causes, hacktivist groups (such as Anonymous) typically execute denial-of-service (DoS) attacks against a target organization by defacing their websites or flooding their networks with traffic.

- **Cyberterrorists.** Terrorist organizations use the internet to recruit, train, instruct, and communicate, and to spread fear and panic to advance their ideologies. Unlike other threat actors, cyberterrorists are largely indiscriminate in their attacks, and their objectives include physical harm, death, and destruction.

Key Terms

Business email compromise (BEC) is the unauthorized use of email leading to financial fraud. BEC techniques including spamming and phishing, among others.

External threat actors – including organized crime, state-affiliated groups, activists, former employees, and other unaffiliated or otherwise unknown attackers – account for the majority of data breaches. Internal actors were involved in approximately 22 percent of reported data breaches, but in some industries – such as financial services (including insurance) – internal actors accounted for as much as 44 percent of breaches.[9]

[8] Zheng, Denise E. "Global Forecast 2016: Disrupting the Cyber Status Quo." Center for Strategic and International Studies. November 16, 2015. https://www.csis.org/analysis/disrupting-cyber-status-quo.

[9] "Verizon 2021 Data Breach Investigations Report." Verizon Enterprise Solutions. Accessed January 16, 2022. https://verizon.com/dbir/.

1.1.2 Modern cyberattack strategy

Modern cyberattack strategy has evolved from a direct attack against a high-value server or asset ("shock and awe") to a patient, multistep process that blends exploits, malware, stealth, and evasion in a coordinated network attack ("low and slow").

The Cyberattack Lifecycle (see Figure 1-1) illustrates the sequence of events that an attacker goes through to infiltrate a network and exfiltrate (or steal) valuable data. Blocking of just one step breaks the chain and can effectively defend an organization's network and data against an attack.

Figure 1-1

The Cyberattack Lifecycle

1. **Reconnaissance.** Attackers meticulously plan their cyberattacks. They research, identify, and select targets, often extracting public information from targeted employees' social media profiles or from corporate websites, which can be useful for social engineering and phishing schemes. Attackers will also use various tools to scan for network vulnerabilities, services, and applications that they can exploit, such as:

    - **Network analyzers** (also known as packet analyzers, protocol analyzers, or packet sniffers) are used to monitor and capture raw network traffic (packets). Examples include tcpdump and Wireshark (formerly Ethereal).

    - **Network vulnerability scanners** typically consist of a suite of tools including password crackers, port scanners, and vulnerability scanners and are used to probe a network for vulnerabilities (including configuration errors) that can be exploited. Examples include Nessus and SAINT.

    - **Password crackers** are used to perform brute-force dictionary attacks against password hashes. Examples include John the Ripper and THC Hydra.

    - **Port scanners** are used to probe for open TCP or UDP (including ICMP) ports on an endpoint. Examples include Nmap ("network mapper") and Nessus.

- **Web application vulnerability scanners** are used to scan web applications for vulnerabilities such as cross-site scripting, SQL injection, and directory traversal. Examples include Burp Suite and OWASP Zed Attack Proxy (ZAP).

- **Wi-Fi vulnerability scanners** are used to scan wireless networks for vulnerabilities (including open and misconfigured access points), to capture wireless network traffic and to crack wireless passwords. Examples include Aircrack-ng and Wifite.

Breaking the Cyberattack Lifecycle at this phase of an attack begins with proactive and effective end-user security awareness training that focuses on topics such as social engineering techniques (such as phishing, piggybacking, and shoulder surfing), social media (such as safety and privacy issues), and organizational security policies (such as password requirements, remote access, and physical security). Another important countermeasure is continuous monitoring and inspection of network traffic flows in order to detect and prevent unauthorized port and vulnerability scans, host sweeps, and other suspicious activity. Effective change and configuration management processes help to ensure that newly deployed applications and endpoints are properly configured (such as disabling unneeded ports and services) and maintained.

2. **Weaponization.** Next, attackers determine which methods to use to compromise a target endpoint. They may choose to embed intruder code within seemingly innocuous files such as a PDF or Microsoft Word document or email message. Or, for highly targeted attacks, attackers may customize deliverables to match the specific interests of an individual within the target organization.

   Breaking the Cyberattack Lifecycle at this phase of an attack is challenging because weaponization typically occurs within the attacker's network. However, analysis of artifacts (both malware and weaponizer) can provide important threat intelligence to enable effective zero-day protection when delivery (the next step) is attempted.

3. **Delivery.** Attackers then attempt to deliver their weaponized payload to a target endpoint; for example, via email, instant messaging (IM), drive-by download (an end user's web browser is redirected to a webpage that automatically downloads malware to the endpoint in the background), or infected file share.

   Breaking the Cyberattack Lifecycle at this phase of an attack requires visibility into all network traffic (including remote and mobile devices) to effectively block malicious or risky websites, applications, and IP addresses, and preventing known and unknown malware and exploits.

4. **Exploitation.** After a weaponized payload is delivered to a target endpoint, it must be triggered. An end user may unwittingly trigger an exploit, for example, by clicking a malicious link or opening an infected attachment in an email, or an attacker may remotely trigger an exploit against a known server vulnerability on the target network.

   As during the Reconnaissance phase, breaking the Cyberattack Lifecycle at this phase of an attack begins with proactive and effective end-user security awareness training that focuses on topics such as malware prevention and email security. Other important security countermeasures include vulnerability and patch management; malware detection and prevention; threat intelligence (including known and unknown threats); blocking risky, unauthorized, or unneeded applications and services; managing file or directory permissions and root or administrator privileges; and logging and monitoring network activity.

5. **Installation.** Next, an attacker will escalate privileges on the compromised endpoint, by, for example, establishing remote shell access and installing rootkits or other malware. With remote shell access, the attacker has control of the endpoint and can execute commands in privileged mode from a command-line interface (CLI) as if physically sitting in front of the endpoint. The attacker will then move laterally across the target's network, executing attack code, identifying other targets of opportunity, and compromising additional endpoints to establish persistence.

   The key to breaking the Cyberattack Lifecycle at this phase of an attack is to limit or restrict the attackers' lateral movement within the network. Use network segmentation and a Zero Trust model that monitors and inspects all traffic between zones or segments, and granular control of applications that are allowed on the network.

6. **Command and Control.** Attackers establish encrypted communication channels back to command-and-control (C2) servers across the internet in order to modify their attack objectives and methods as additional targets of opportunity are identified within the victim network, or to evade any new security countermeasures that the organization may attempt to deploy if attack artifacts are discovered. Communication is essential to an attack because it enables the attacker to remotely direct the attack and execute attack objectives. C2 traffic must therefore be resilient and stealthy for an attack to succeed. Attack communication traffic is usually hidden with various techniques and tools, including:

   - **Encryption** with SSL, SSH (Secure Shell), or some other custom or proprietary encryption.

- **Circumvention** via proxies, remote access tools, or tunneling. In some instances, use of cellular networks enables complete circumvention of the target network for attack C2 traffic.

- **Port evasion** using network anonymizers or port hopping to traverse over any available open ports.

- **Fast Flux (or Dynamic DNS)** to proxy through multiple infected endpoints or multiple, ever-changing C2 servers in order to reroute traffic and make determination of the true destination or attack source difficult.

- **DNS tunneling** is used for C2 communications, as well as data infiltration (for example, by sending malicious code, commands, or binary files to a victim) and data exfiltration.

Breaking the Cyberattack Lifecycle at this phase of an attack requires inspection of all network traffic (including encrypted communications), blocking of outbound C2 communications with anti-C2 signatures (along with file and data pattern uploads), blocking of all outbound communications to known malicious URLs and IP addresses, blocking of novel attack techniques that employ port evasion methods, prevention of the use of anonymizers and proxies on the network, monitoring of DNS for malicious domains and countering with DNS sinkholing or DNS poisoning, and the redirection of malicious outbound communications to honeypots to identify or block compromised endpoints and analyze attack traffic.

7. **Actions on the Objective.** Attackers often have multiple, different attack objectives, including data theft; destruction or modification of critical systems, networks, and data; and denial-of-service (DoS). This last stage of the Cyberattack Lifecycle can also be used by an attacker to advance the early stages of the Cyberattack Lifecycle against another target. The Verizon *2021 Data Breach Investigations Report* (DBIR) describes this strategy as a secondary motive in which "the ultimate goal of an incident was to leverage the victim's access, infrastructure or any other asset to conduct other incidents."[10] For example, an attacker may compromise a company's extranet to breach a business partner that is the primary target. According to the DBIR, in 2020 there were 24,913 incidents in which "web apps were attacked with a secondary motive."[11] The

[10] "Verizon 2021 Data Breach Investigations Report." Verizon Enterprise Solutions. Accessed January 16, 2022. https://verizon.com/dbir/.

[11] Ibid.

attacker pivots the attack against the initial victim network to a different victim network, thus making the initial victim an unwitting accomplice.

1.1.3 MITRE ATT&CK framework

The MITRE Adversarial Tactics, Techniques, and Common Knowledge (ATT&CK) framework is a comprehensive matrix of tactics and techniques designed for threat hunters, defenders, and red teams to help classify attacks, identify attack attribution and objective, and assess an organization's risk. Organizations can use the framework to identify security gaps and prioritize mitigations based on risk.

MITRE started ATT&CK in 2013 to document the TTPs that advanced persistent threats (APTs) use against enterprise networks. It was created out of a need to describe adversary TTPs that would be used by a MITRE research project called FMX. The objective of FMX was to investigate how endpoint telemetry data and analytics could help improve post-intrusion detection of attackers operating within enterprise networks. The ATT&CK framework was used as the basis for testing the efficacy of the sensors and analytics under FMX and served as the common language both offense and defense could use to improve over time.

MITRE ATT&CK now has three iterations:

- **ATT&CK for Enterprise:** Focuses on adversarial behavior in Windows, Mac, Linux, and cloud environments.
- **ATT&CK for Mobile:** Focuses on adversarial behavior on iOS and Android operating systems.
- **Pre-ATT&CK:** Focuses on “pre-exploit” adversarial behavior. Pre-ATT&CK is included as part of the ATT&CK for Enterprise matrix.

Techniques represent “how” an adversary achieves a tactical goal by performing an action. For example, an adversary may dump credentials to achieve credential access. The MITRE ATT&CK matrix contains a set of techniques used by adversaries to accomplish a specific objective. Those objectives are categorized as tactics in the ATT&CK Matrix. The Enterprise ATT&CK matrix is a superset of the Windows, MacOS, and Linux matrices. MITRE regularly updates the techniques discovered in the wild by both cybersecurity researchers and hackers alike. As of 2022, there are 218 techniques defined in the Enterprise model.

Sub-techniques are a more specific description of the adversarial behavior used to achieve a goal. They describe behavior at a lower level than a technique. For example, an adversary may dump credentials by accessing the Local Security Authority (LSA) Secrets.

Tactics represent the “why” of an ATT&CK technique or sub-technique. Adversarial tactics represent the attacker's goal or the reason for performing an action. For example, an adversary may want to achieve credential access.

Tactics are listed in Table 1-1.

Table 1-1

MITRE Tactics

Tactic	The attacker is trying to:
Reconnaissance	Gather information they can use to plan future operations
Resource Development	Establish resources they can use to support operations
Initial Access	Get into your network
Execution	Run malicious code
Persistence	Maintain their foothold
Privilege Escalation	Gain higher-level permissions
Defense Evasion	Avoid being detected
Credential Access	Steal account names and passwords
Discovery	Figure out your environment
Lateral Movement	Move through your environment
Collection	Gather data of interest to their goal
Command and Control	Communicate with compromised systems to control them
Exfiltration	Steal data
Impact	Manipulate, interrupt, or destroy your systems and data

Procedures are the specific implementation the adversary uses for techniques or sub-techniques. For example, a procedure could be an adversary using PowerShell to inject into lsass.exe to dump credentials by scraping LSASS memory on a victim. Procedures are categorized in the ATT&CK framework as techniques observed in the wild in the “Procedure Examples" section of technique pages.

Sub-techniques and procedures describe different things in ATT&CK. Sub-techniques are used to categorize behavior and procedures are used to describe in-the-wild use of techniques. Furthermore, since procedures are specific implementations of techniques and sub-techniques,

they may include several additional behaviors in how they are performed. For example, an adversary using PowerShell to inject into lsass.exe to dump credentials by scraping LSASS memory on a victim is a procedure implementation containing several (sub)techniques covering PowerShell, Credential Dumping and Process Injection used against LSASS.

1.1 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **True or False.** Most cyberattacks today are perpetrated by internal threat actors such as malicious employees engaging in corporate espionage.
2. **Classroom Discussion.** Describe the different motivations of various adversaries, including cybercriminals, cyberterrorists, state-sponsored organizations, and hacktivists.
3. **True or False.** The Cyberattack Lifecycle is a five-step process that an attacker follows to attack a network.
4. **Multiple Answer.** List and describe the steps of the Cyberattack Lifecycle.
5. **True or False.** An attacker needs to succeed in executing only one step of the Cyberattack Lifecycle to infiltrate a network, whereas a defender must "be right every time" and break every step of the chain to prevent an attack.
6. **Multiple Choice.** Which technique is not used to break the command-and-control (C2) phase of the Cyberattack Lifecycle? (Choose one.)
    a) blocking outbound traffic to known malicious sites and IP addresses
    b) DNS sinkholing and DNS poisoning
    c) vulnerability and patch management
    d) all of the above
7. **True or False.** The key to breaking the Cyberattack Lifecycle during the Installation phase is to implement network segmentation, a Zero Trust model, and granular control of applications to limit or restrict an attacker's lateral movement within the network.

1.2 Cyberattack Techniques and Types

Attackers use a variety of techniques and attack types to achieve their objectives. *Malware* and *exploits* are integral to the modern cyberattack strategy. Spamming and phishing are commonly employed techniques to deliver malware and exploits to an endpoint via an email executable or a web link to a malicious website. After an endpoint is compromised, an attacker typically installs backdoors, remote access Trojans, and other malware to ensure persistence. Compromised endpoints ("bots") under the control of an attacker are often used to perpetrate much larger-scale attacks against other organizations or networks as part of a botnet. This section describes different types of malware, vulnerabilities, and exploits; business email compromise (BEC); and how bots and botnets function, along with different types of botnets.

Key Terms

Malware is malicious software or code that typically takes control of, collects information from, or damages an infected endpoint. Malware broadly includes viruses, worms, Trojan horses (including remote access Trojans, or RATs), ransomware, anti-AV, logic bombs, backdoors, rootkits, bootkits, spyware, and (to a lesser extent) adware.

An *exploit* is a small piece of software code, part of a malformed data file, or a sequence (string) of commands that leverages a vulnerability in a system or software, causing unintended or unanticipated behavior in the system or software.

A *vulnerability* is a bug or flaw that exists in a system or software and creates a security risk.

1.2.1 Malware and ransomware

Malware is malicious software or code that typically takes control of, collects information from, or damages an infected endpoint. Malware broadly includes:

- **Viruses.** A virus is malware that is self-replicating but must first infect a host program and be executed by a user or process.
- **Worms.** A worm is malware that typically targets a computer network by replicating itself to spread rapidly. Unlike viruses, worms do not need to infect other programs and do not need to be executed by a user or process.
- **Trojan horses.** A Trojan horse is malware that is disguised as a harmless program but actually gives an attacker full control and elevated privileges of an endpoint when installed. Unlike other types of malware, Trojan horses are typically not self-replicating.

- **Ransomware.** Ransomware is malware that locks a computer or device (locker ransomware) or encrypts data (crypto ransomware) on an infected endpoint with an encryption key that only the attacker knows, thereby making the data unusable until the victim pays a ransom (usually in cryptocurrency such as Bitcoin). Reveton and LockeR are two examples of locker ransomware, while Locky, TeslaCrypt/EccKrypt, Cryptolocker, and Cryptowall are examples of crypto ransomware.

- **Anti-AV.** Anti-AV is malware that disables legitimately installed antivirus software on the compromised endpoint, thereby preventing automatic detection and removal of other malware.

- **Logic bombs.** A logic bomb is malware that is triggered by a specified condition, such as a given date or a particular user account being disabled.

- **Backdoors.** A backdoor is malware that allows an attacker to bypass authentication to gain access to a compromised system.

- **Rootkits.** A rootkit is malware that provides privileged (root-level) access to a computer. Rootkits are installed in the BIOS of a machine, which means operating system-level security tools cannot detect them.

- **Bootkits.** A bootkit is malware that is a kernel-mode variant of a rootkit, commonly used to attack computers that are protected by full-disk encryption.

- **Spyware and adware.** Spyware and adware are types of malware that collect information, such as internet surfing behavior, login credentials, and financial account information on an infected endpoint. Spyware often changes browser and other software settings, and slows computer and internet speeds on an infected endpoint. Adware is spyware that displays annoying advertisements on an infected endpoint, often as pop-up banners.

Early malware typically consisted of viruses that displayed annoying – but relatively benign – errors, messages, or graphics.

The first computer virus was Elk Cloner, written in 1982 by a ninth-grade high school student near Pittsburgh, Pennsylvania. Elk Cloner was a relatively benign *boot sector* virus that displayed a poem on the fiftieth time that an infected *floppy disk* was inserted into an Apple II computer.

The first PC virus was a boot sector virus, written in 1986, called Brain. Brain was also relatively benign and displayed a message with the actual contact information for the creators of the

virus. Brain was written by two Pakistani brothers who created the virus so that they could track piracy of their medical software.

Key Terms

A *boot sector virus* targets the boot sector or *master boot record* (MBR) of an endpoint's storage drive or other removable storage media.

A *boot sector* contains machine code that is loaded into an endpoint's memory by firmware during the startup process, before the operating system is loaded.

A *master boot record* (MBR) contains information about how the logical partitions (or file systems) are organized on the storage media and an executable boot loader that starts up the installed operating system.

A *floppy disk* is a removable magnetic storage medium commonly used from the mid-1970s until about 2007, when it was largely replaced by compact discs and removable USB storage devices. Floppy disks were typically available in 8-inch, 5¼-inch, and 3½-inch sizes with capacities from 90 kilobytes to 200 megabytes.

One of the first computer worms to gain widespread notoriety was the Morris worm, written by a Harvard and Cornell University graduate student, Robert Tappan Morris, in 1988. The worm exploited weak passwords and known vulnerabilities in several Unix programs and spread rapidly across the early internet (the worm infected up to an estimated 10 percent of all Unix machines connected to the internet at that time – about 6,000 computers), sometimes infecting a computer numerous times to the point that it was rendered useless – an example of an early DoS attack. The U.S. Government Accountability Office (GAO) estimated the damage caused by the Morris worm between US$100,000 and US$10 million.

Unfortunately, more than 35 years since these early examples of malware, modern malware has evolved and is used for far more sinister purposes. Examples of modern malware include:

- **WannaCry.** In a period of just 24 hours in May 2017, the WannaCry ransomware attack infected more than 230,000 vulnerable Windows computers in more than 150 countries worldwide. Although the attack was quickly halted after the discovery of a "kill switch," the total economic damage is estimated between hundreds of millions to as much as US$4 billion, despite the perpetrators collecting only 327 ransom payments totaling about US$130,000.
- **HenBox.** HenBox typically masquerades as legitimate Android system and VPN apps, and sometimes drops and installs legitimate versions of other apps as a decoy. The

primary goal of the HenBox apps appears to be to spy on those who install them. By using traits similar to legitimate apps, for example, copycat iconography and app or package names, HenBox lures victims into downloading and installing the malicious apps from third-party, non-Google Play app stores that often have fewer security and vetting procedures for the apps they host. As with other Android malware, some apps may also be available on forums or file-sharing sites, or even may be sent to victims as email attachments.

- **TeleRAT.** Telegram Bots are special accounts that do not require an additional phone number to set up and are generally used to enrich Telegram chats with content from external services or to get customized notifications and news. TeleRAT abuses Telegram's Bot API for C2 and data exfiltration.

- **Rarog.** Rarog is a cryptocurrency-mining Trojan that has been sold on various underground forums since June 2017 and has been used by countless criminals since then. Rarog has been primarily used to mine the Monero cryptocurrency, but it can mine others. It comes equipped with several features, including providing mining statistics to users, configuring various processor loads for the running miner, the ability to infect USB devices, and the ability to load additional *dynamic-link libraries* (DLLs) on the victim device. Rarog provides an affordable way for new criminals to gain entry using this particular type of malware. Other examples of cryptocurrency miners include Coinhive, JSE-Coin, Crypto-Loot, and CoinImp.

Key Terms

A *dynamic-link library* (DLL) is a type of file used in Microsoft operating systems that enables multiple programs to simultaneously share programming instructions contained in a single file to perform specific functions.

Modern malware is typically stealthy and evasive, and now plays a central role in a coordinated attack against a target (see Section 1.1.2).

Advanced malware leverages networks to gain power and resilience, and can be updated – just like any other software application – so that an attacker can change course and dig deeper into the network or make changes and enact countermeasures.

This is a fundamental shift compared to earlier types of malware, which were generally independent agents that simply infected and replicated themselves. Advanced malware increasingly has become a centrally coordinated, networked application in a very real sense. In much the same way that the internet changed what was possible in personal computing,

ubiquitous network access is changing what is possible in the world of malware. Now all malware of the same type can work together toward a common goal, with each infected endpoint expanding the attack foothold and increasing the potential damage to the organization.

Important characteristics and capabilities of advanced malware include:

- **Distributed, fault-tolerant architecture.** Advanced malware takes full advantage of the resiliency built into the internet itself. Advanced malware can have multiple control servers distributed all over the world with multiple fallback options, and can also leverage other infected endpoints as communication channels, thus providing a near infinite number of communication paths to adapt to changing conditions or update code as needed.

- **Multifunctionality.** Updates from C2 servers can also completely change the functionality of advanced malware. This multifunctional capability enables an attacker to use various endpoints strategically to accomplish specific desired tasks, such as stealing credit card numbers, sending spam containing other malware payloads (such as spyware), or installing ransomware for the purpose of extortion.

- **Polymorphism and metamorphism.** Some advanced malware has entire sections of code that serve no purpose other than to change the signature of the malware, thus producing an infinite number of unique signature hashes for even the smallest of malware programs. Techniques such as *polymorphism* and *metamorphism* are used to avoid detection by traditional signature-based anti-malware tools and software. For example, a change of just a single character or bit of the file or source code completely changes the *hash signature* of the malware.

- **Obfuscation.** Advanced malware often uses common *obfuscation* techniques to hide certain binary strings that are characteristically used in malware and therefore are easily detected by anti-malware signatures, or to hide an entire malware program.

Although ransomware is technically classified as malware, the surge in ransomware attacks over the past five years warrants additional consideration. Ransomware is a criminal business model that uses malware to hold something of value for ransom. Victims of a ransomware attack may have their operations severely degraded or shut down entirely. Although cryptographic ransomware is the most common and successful type of ransomware, it is not the only one. It's important to remember that ransomware is not a single family of malware but is a criminal business model in which malware is used to hold something of value for ransom.

While holding something of value for ransom is not a new concept, ransomware has become a multibillion-dollar criminal business targeting both individuals and corporations. Due to its low barriers to entry and effectiveness in generating revenue, it has quickly displaced other cybercrime business models and become the largest threat facing organizations today. It is also important to note that although threat actors generally do decrypt your data after the ransom is paid (the ransomware business model depends on a reasonable expectation that paying a ransom will restore access to your data), there are no guarantees that this will be the case. Additionally, many threat actors are now exfiltrating a copy of their victims' data – particularly PII and credit card numbers – before encrypting it, then selling the data on the dark web after the ransom is paid.

For a ransomware attack to be successful, attackers must execute the following five steps:

1. **Compromise and control a system or device.** Ransomware attacks typically begin by using social engineering to trick users into opening an attachment or viewing a malicious link in their web browser. This allows attackers to install malware onto a system and take control. However, another increasingly common tactic is for attackers to gain access to the network, perform reconnaissance on the network to identify potential targets and establish C2, install other malware and create backdoor accounts for persistence, and potentially exfiltrate data.

2. **Prevent access to the system.** Attackers will either identify and encrypt certain file types or deny access to the entire system.

3. **Notify victim.** Though seemingly obvious, attackers and victims often speak different languages and have varying levels of technical capabilities. Attackers must alert the victim about the compromise, state the demanded ransom amount, and explain the steps for regaining access.

4. **Accept ransom payment.** To receive payment while evading law enforcement, attackers utilize cryptocurrencies such as Bitcoin for the transaction.

5. **Return full access.** Attackers must return access to the device(s). Failure to restore the compromised systems destroys the effectiveness of the scheme as no one would be willing to pay a ransom if they didn't believe their valuables would be returned.

If the attacker fails in any of these steps, the scheme will be unsuccessful. Although the concept of ransomware has existed for decades, the technology and techniques, such as reliable encrypting and decrypting, required to complete all five of these steps on a wide scale were not available until just a few years ago.

Though the malware deployed in the current generation of cryptographic ransomware attacks is not especially sophisticated, it has proven very effective at not only generating revenue for the criminal operators but also preventing impacted organizations from continuing their normal operations. New headlines each week demonstrate that organizations large and small are vulnerable to these threats, enticing new attackers to jump onto the bandwagon and begin launching their own ransomware campaigns.

Key Terms

Polymorphism alters part of the malware code with every iteration, such as the encryption key or decryption routine, but the malware payload remains unchanged.

Metamorphism uses more advanced techniques than polymorphism to alter malware code with each iteration. Although the malware payload changes with each iteration – for example, by using a different code structure or sequence or by inserting garbage code to change the file size – the fundamental behavior of the malware payload remains unchanged.

A *hash signature* is a cryptographic representation of an entire file or program's source code.

Obfuscation is a programming technique used to render code unreadable. It can be implemented by using a simple substitution cipher, such as an *exclusive or* (XOR) operation – in which the output is true only when the inputs are different (for example, TRUE and TRUE equals FALSE, but TRUE and FALSE equals TRUE) – or by using more sophisticated encryption algorithms, such as the *Advanced Encryption Standard* (AES). Alternatively, a *packer* can be used to compress a malware program for delivery and then decompress it in memory at runtime.

1.2.2 Vulnerabilities and exploits

An exploit is a type of malware that takes advantage of a vulnerability in installed endpoint or server software such as a web browser, Adobe Flash, Java, or Microsoft Office. An attacker crafts an exploit that targets a software vulnerability, causing the software to perform functions or execute code on behalf of the attacker.

Vulnerabilities are routinely discovered in software at an alarming rate. Vulnerabilities may exist in software when the software is initially developed and released, or vulnerabilities may be inadvertently created, or even reintroduced, when subsequent version updates or security patches are installed. According to research by Palo Alto Networks, 78 percent of exploits take advantage of vulnerabilities that are less than two years old.

Security patches are developed by software vendors as quickly as possible after a vulnerability has been discovered in their software. However, an attacker may learn of a vulnerability and begin exploiting it before the software vendor is aware of the vulnerability or has an opportunity to develop a patch. This delay between the discovery of a vulnerability and development and release of a patch is known as a zero-day threat (or exploit). It may be months or years before a vulnerability is announced publicly. After a security patch becomes available, time inevitably is required for organizations to properly test and deploy the patch on all affected systems. During this time, a system running the vulnerable software is at risk of being exploited by an attacker (see Figure 1-2).

Figure 1-2

Vulnerabilities can be exploited from the time software is deployed until it is patched.

Exploits can be embedded in seemingly innocuous data files (such as Microsoft Word documents, PDF files, and webpages), or they can target vulnerable network services. Exploits are particularly dangerous because they are often packaged in legitimate files that do not trigger anti-malware (or antivirus) software and are therefore not easily detected.

Creation of an exploit data file is a two-step process. The first step is to embed a small piece of malicious code within the data file. However, the attacker still must trick the application into running the malicious code. Thus, the second part of the exploit typically involves memory corruption techniques that allow the attacker's code to be inserted into the execution flow of the vulnerable software. After that happens, a legitimate application, such as a document viewer or web browser, will perform actions on behalf of the attacker, such as establishing communication and providing the ability to upload additional malware to the target endpoint. Because the application being exploited is a legitimate application, traditional signature-based antivirus and whitelisting software have virtually no effectiveness against these attacks.

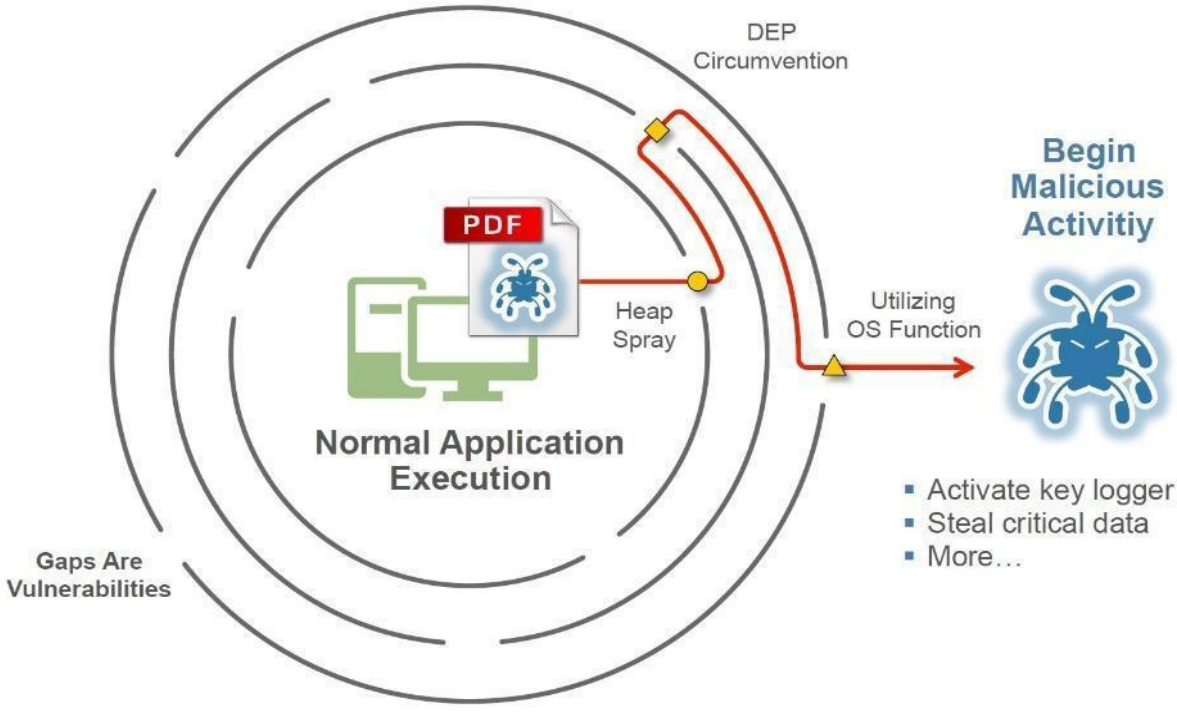
Although there are many thousands of exploits, they all rely on a small set of core techniques that change infrequently. For example, a *heap spray* is an attempt to insert the attacker's code into multiple locations within the memory heap, hoping that one of those locations will be called by the process and executed. Some attacks may involve more steps, some may involve fewer, but typically three to five core techniques must be used to exploit an application. Regardless of the attack or its complexity, for the attack to be successful the attacker must execute a series of these core exploit techniques in sequence, like navigating a maze to reach its objective (see Figure 1-3).

Key Terms

Heap spray is a technique used to facilitate arbitrary code execution by injecting a certain sequence of bytes into the memory of a target process.

Figure 1-3

Exploits rely on a series of core attack techniques to succeed.



1.2.3 Business email compromise (BEC)

Business email compromise (BEC) is one of the most prevalent types of cyberattacks that organizations face today. The FBI Internet Crime Complaint Center (IC3) estimates that in

aggregate, BEC attacks cost organizations three times more than any other cybercrime[12] and BEC incidents represented nearly a third of the incidents investigated by the Palo Alto Networks Unit 42 Incident Response Team in 2021. According to the Verizon *2021 Data Breach Investigations Report (DBIR)*, BEC is the second most common form of social engineering today[13].

Spam and phishing emails are the most common delivery methods for malware. The volume of spam email as a percentage of total global email traffic fluctuates widely from month to month – typically 45 to 75 percent. Although most end users today are readily able to identify spam emails and are savvier about not clicking links, opening attachments, or replying to spam emails, spam remains a popular and effective infection vector for the spread of malware.

Phishing attacks, in contrast to spam, are becoming more sophisticated and difficult to identify.

Spear phishing is a targeted phishing campaign that appears more credible to its victims by gathering specific information about the target and thus has a higher probability of success. A spear phishing email may spoof an organization (such as a financial institution) or individual that the recipient actually knows and does business with, and it may contain very specific information (such as the recipient's first name, rather than just an email address).

Whaling is a type of spear phishing attack that is specifically directed at senior executives or other high-profile targets within an organization. A whaling email typically purports to be a legal subpoena, customer complaint, or other serious matter.

Spear phishing, and phishing attacks in general, is not always conducted via email. A link is all that is required, such as a link on Facebook or on a message board, or a shortened URL on Twitter. These methods are particularly effective in spear phishing attacks because they allow the attacker to gather a great deal of information about the targets and then lure them through dangerous links into a place where the users feel comfortable.

Watering hole attacks compromise websites that are likely to be visited by a targeted victim, for example, an insurance company website that may be frequently visited by healthcare providers. The compromised website will typically infect unsuspecting visitors with malware (known as a "drive-by download").

---

[12] "2020 Internet Crime Report." Federal Bureau of Investigation. Accessed January 16, 2022. https://www.ic3.gov/Media/PDF/AnnualReport/2020_IC3Report.pdf.

[13] "Verizon 2021 Data Breach Investigations Report." Verizon Enterprise Solutions. Accessed January 16, 2022. https://verizon.com/dbir/.

A *pharming* attack redirects a legitimate website's traffic to a fake site, typically by modifying an endpoint's local hosts file or by compromising a DNS server ("DNS poisoning").

Key Terms

Spear phishing is a highly targeted phishing attack that uses specific information about the target to make the phishing attempt appear legitimate.

Whaling is a type of spear phishing attack that is specifically directed at senior executives or other high-profile targets within an organization.

Watering hole attacks compromise websites that are likely to be visited by a targeted victim to deliver malware via a drive-by download. A *drive-by download* is a software download, typically malware, that happens without a user's knowledge or permission.

Pharming is a type of attack that redirects a legitimate website's traffic to a fake site.

1.2.4 Bots and botnets

Bots and *botnets* are notoriously difficult for organizations to detect and defend against using traditional anti-malware solutions.

Key Terms

Bots (or *zombies*) are individual endpoints that are infected with advanced malware that enables an attacker to take control of the compromised endpoint.

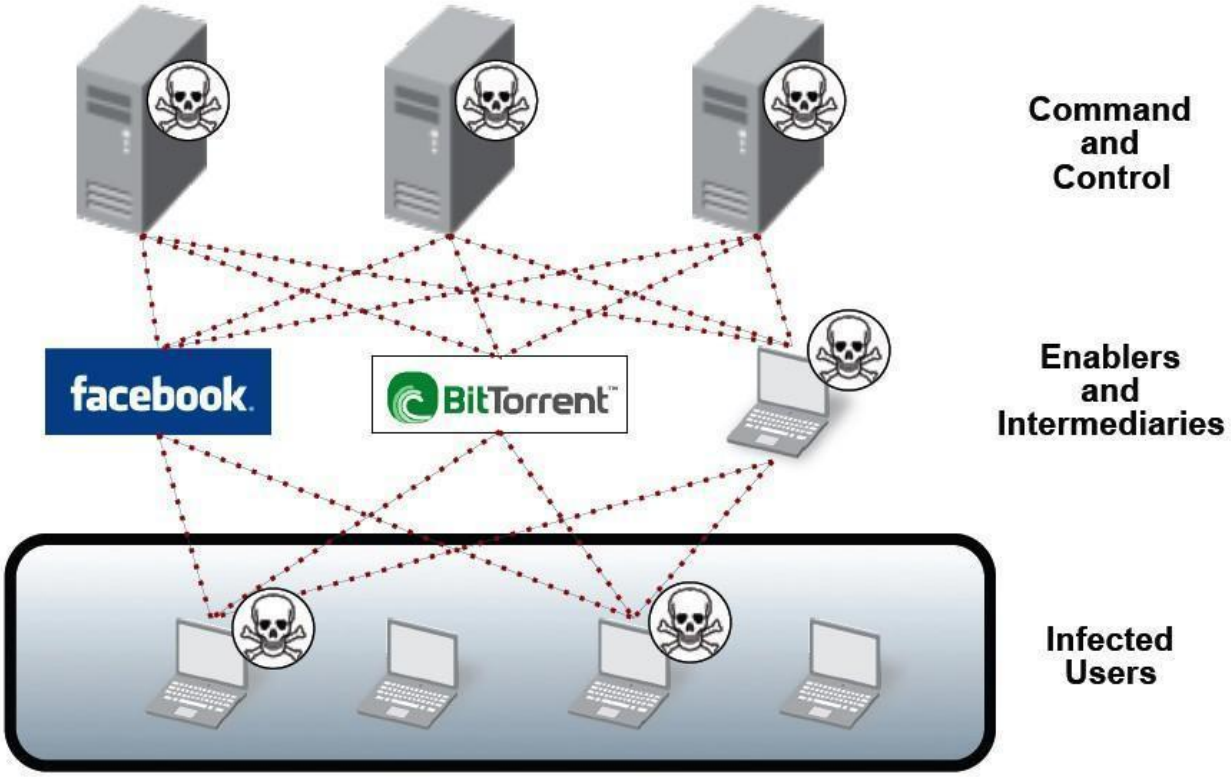
A *botnet* is a network of bots (often tens of thousands or more) working together under the control of attackers using numerous servers.

In a botnet, advanced malware works together toward a common objective, with each bot growing the power and destructiveness of the overall botnet. The botnet can evolve to pursue new goals or adapt as different security countermeasures are deployed. Communication between the individual bots and the larger botnet through C2 servers provides resiliency in the botnet (see Figure 1-4).

Given their flexibility and ability to evade defenses, botnets present a significant threat to organizations. The ultimate impact of a botnet is largely left up to the attacker, from sending spam one day to stealing credit card data the next – and far beyond, because many cyberattacks go undetected for months or even years.

Figure 1-4

The distributed C2 infrastructure of a botnet



Botnets themselves are dubious sources of income for cybercriminals. Botnets are created by cybercriminals to harvest computing resources (bots). Control of botnets (through C2 servers) can then be sold or rented out to other cybercriminals.

The key to “taking down” or “decapitating” a botnet is to separate the bots (infected endpoints) from their brains (C2 servers). If the bots cannot get to their servers, they cannot get new instructions, upload stolen data, or do anything that makes botnets so unique and dangerous.

Although this approach may seem straightforward, extensive resources are typically required to map the distributed C2 infrastructure of a botnet, and this approach almost always requires an enormous amount of investigation, expertise, and coordination between numerous industry, security, and law enforcement organizations worldwide.

The disabling of C2 servers often requires both physically seizing the servers and taking ownership of the domain and/or IP address range associated with the servers. Very close coordination between technical teams, legal teams, and law enforcement is essential to disabling the C2 infrastructure of a botnet. Many botnets have C2 servers all over the world and will specifically function in countries that have little or no law enforcement for internet crimes.

Further complicating takedown efforts is the fact that a botnet almost never relies on a single C2 server but rather uses multiple C2 servers for redundancy purposes. Each server also is typically insulated by a variety of intermediaries to cloak the true location of the server. These intermediaries include P2P networks, blogs, and social networking sites, and even communications that proxy through other infected bots. These evasion techniques make simply finding C2 servers a considerable challenge.

Most botnets are also designed to withstand the loss of a C2 server, meaning that the *entire* botnet C2 infrastructure must be disabled almost simultaneously. If any C2 server is accessible or any of the fallback options survive, the bots will be able to get updates and rapidly populate a completely new set of C2 servers, and the botnet will quickly recover. Thus, even a single C2 server remaining functional for even a small amount of time can give an attacker the window needed to update the bots and recover the entire botnet.

Spamhaus Malware Labs identified 5,778 botnet C2 servers in the first three quarters of 2021.[14] Botnet C2 servers are used to control infected endpoints (bots) and to exfiltrate personal and/or valuable data from bots. Botnets can be easily scaled up to send massive volumes of spam, spread ransomware, launch *distributed denial-of-service* (DDoS) attacks, commit click-fraud campaigns, and/or mine cryptocurrency (such as Bitcoin).

Key Terms

Distributed denial-of-service (DDOS) is a type of cyberattack in which extremely high volumes of network traffic such as packets, data, or transactions are sent to the target victim's network to make their network and systems (such as an e-commerce website or other web application) unavailable or unusable.

1.2.4.1 Spamming botnets

The largest botnets are often dedicated to sending spam. The premise is straightforward: The attacker attempts to infect as many endpoints as possible, and the endpoints can then be used to send out spam email messages without the end users' knowledge. The relative impact of this type of bot on an organization may seem low initially, but an infected endpoint sending spam could consume additional bandwidth and ultimately reduce the productivity of the users and even the network itself. Perhaps more consequential is the fact that the organization's email domain and IP addresses could also easily become listed by various real-time blackhole lists

[14] "Spamhaus Botnet Threat Update Q3 2021." Spamhaus Malware Labs. Accessed January 16, 2022. https://www.spamhaus.org/news/images/botnet-report-2021-q3/spamhaus-botnet-report-2021-q3.pdf.

(RBLs), causing legitimate emails to be labeled as spam and blocked by other organizations, and damaging the reputation of the organization.

The Rustock botnet is an example of a spamming botnet. It could send up to 25,000 spam email messages per hour from an individual bot and, at its peak, sent an average of 192 spam emails per minute per bot. Rustock is estimated to have infected more than 2.4 million computers worldwide. In March 2011, the U.S. Federal Bureau of Investigation (FBI), working with Microsoft and others, was able to take down the Rustock botnet, which had operated for more than five years and at the time was responsible for sending up to 60 percent of the world's spam.

1.2.4.2 Distributed denial-of-service botnets

A DDoS attack is a type of cyberattack in which extremely high volumes of network traffic such as packets, data, or transactions are sent to the target victim's network to make their network and systems (such as an e-commerce website or other web application) unavailable or unusable. A DDoS botnet uses bots as part of a DDoS attack, overwhelming a target server or network with traffic from a large number of bots. In such attacks, the bots themselves are not the target of the attack. Instead, the bots are used to flood some other remote target with traffic. The attacker leverages the massive scale of the botnet to generate traffic that overwhelms the network and server resources of the target.

Unlike other types of cyberattacks, a DDoS attack does not typically employ a prolonged, stealthy approach. Instead, a DDoS attack more often takes the form of a highly visible brute-force attack that is intended to rapidly cause damage to the victim's network and systems infrastructure and to their business and reputation.

DDoS attacks often target specific organizations for personal or political reasons, or to extort a ransom payment in exchange for stopping the DDoS attack. DDoS attacks are often used by hacktivists (discussed in Section 1.1.1) to promote or protest a particular political agenda or social cause. DDoS attacks may also be used for criminal extortion purposes to extract a hefty ransom payment in exchange for ending the attack.

DDoS botnets represent a dual risk for organizations: The organization itself can be the target of a DDoS attack. And even if the organization isn't the ultimate target, any infected endpoints participating in the attack will consume valuable network resources and facilitate a criminal act, albeit unwittingly.

A DDoS attack can also be used as part of a targeted strategy for a later attack. While the victim organization is busy defending against the DDoS attack and restoring the network and systems, the attacker can deliver an exploit to the victim network (for example, by causing a buffer

overflow in a SQL database) that will enable a malware infection and establish a foothold in the network. The attacker can then return later to expand the (stealthy) attack and extract stolen data.

Examples of recent DDoS attacks include attacks against *World of Warcraft Classic* and Wikipedia in September 2019.[15]

1.2.4.3 Financial botnets

Financial botnets, such as ZeuS and SpyEye, are responsible for the direct theft of funds from all types of enterprises. These types of botnets are typically not as large as spamming or DDoS botnets, which grow as large as possible for a single attacker. Instead, financial botnets are often sold as kits that allow attackers to license the code and build their own botnets.

The impact of a financial breach can be enormous, including the breach of sensitive consumer and financial information, leading to significant financial, legal, and brand damage. As reported by Tech Republic:

"A Mirai botnet variant was used in attacks against at least one financial sector company in January 2018 – possibly the first time an IoT botnet has been observed in use in a DDoS attack since the Mirai botnet took down multiple websites in 2017, according to a Thursday report from Recorded Future."[16]

1.2.5 Advanced persistent threats

Advanced persistent threats (APTs) are a class of threats that are far more deliberate and potentially devastating than other types of cyberattacks. As its name implies, an APT has three defining characteristics. An APT is:

- **Advanced.** Attackers use advanced malware and exploits and typically also have the skills and resources necessary to develop additional cyberattack tools and techniques, and may have access to sophisticated electronic surveillance equipment, satellite imagery, and even human intelligence assets.

---

[15] Oleg Kuprev, Ekaterina Badovskaya, and Alexander Gutnikov. "DDoS attacks in Q3 2019." Kaspersky. November 11, 2019. https://securelist.com/ddos-report-q3-2019/94958/.

[16] Rayome, Alison DeNisco. "Mirai variant botnet launches IoT DDoS attacks on financial sector." Tech Republic. April 5, 2018. https://www.techrepublic.com/article/mirai-variant-botnet-launches-iot-ddos-attacks-on-financial-sector/.

- **Persistent.** An APT may take place over a period of several years. The attackers pursue specific objectives and use a "low-and-slow" approach to avoid detection. The attackers are well organized and typically have access to substantial financial backing, such as a nation-state or organized criminal organization, to fund their activities.

- **Threat.** An APT is deliberate and focused, rather than opportunistic. APTs are designed to cause real damage, including significant financial loss, destruction of systems and infrastructure, and physical harm and loss of life.

Some recent APT threat actors include:

- **Lazarus** (also known as APT38, Gods Apostles, Gods Disciples, Guardians of Peace, ZINC, Whois Team, and Hidden Cobra).[17] The Lazarus APT group is a threat actor linked to North Korea and believed to be behind attacks against more than 16 organizations in at least 11 countries, including the Bangladesh cyber heist (US$81 million was surreptitiously transferred from the New York Federal Reserve Bank account of Bangladesh in February 2016),[18] the Troy Operation (attacks against South Korean infrastructure in 2013),[19] the DarkSeoul Operation (malware-based attacks that wiped tens of thousands of hard drives belonging to South Korean television networks and banks in March 2013),[20] and the Sony Picture hack (employees' emails and personal information including salaries, addresses, and Social Security numbers were revealed, unreleased movies posted on file sharing sites, and internal computer systems shut down in 2014).[21]

[17] "Top 25 Threat Actors – 2019 Edition." SBS CyberSecurity. December 12, 2019. https://sbscyber.com/resources/top-25-threat-actors-2019-edition.

[18] Paganini, Pierluigi. "US blames North Korea for the $81 million Bangladesh cyber heist." Security Affairs. March 24, 2017. http://securityaffairs.co/wordpress/57396/cyber-crime/bangladesh-cyber-heist.html.

[19] Paganini, Pierluigi. "Hackers hit South Korea also spread spyware to steal military secrets." Security Affairs. July 9, 2013. http://securityaffairs.co/wordpress/16014/hacking/hackers-hit-south-korea-spyware-steal-military-secrets.html.

[20] Ibid.

[21] Weisman, Aly. "A Timeline of the Crazy Events in the Sony Hacking Scandal." Business Insider. December 9, 2014. http://www.businessinsider.com/sony-cyber-hack-timeline-2014-12.

- **Fancy Bear** (also known as APT28, Sofacy, Sednit, and Tsar Team).[22,23] Fancy Bear is a Russia-based APT threat actor that has been operating since 2010. Recent targets and attacks have included the German Think Tank Attacks (2019), German elections (2017), World Anti-Doping Agency attack (2016), U.S. Democratic National Committee breach (2016), and Operation "Pawn Storm" (2014).[24]

- **MONSOON** (also known as Patchwork, APT -C-09, Chinastrats, Dropping Elephant, and Quilted Tiger).[25] MONSOON is an APT threat actor that appears to have begun in 2014. According to Forcepoint Security Labs, "The overarching campaign appears to target both Chinese nationals within different industries and government agencies in Southern Asia."[26] As of July 2016, more than 110 different victim countries and 6,300 victim IP addresses had been identified.[27] "The malware components used in MONSOON are typically distributed through [weaponized] documents sent through e-mail to specifically chosen targets. Themes of these documents are usually political in nature and taken from recent publications on topical current affairs. Several malware components have been used in this operation including Unknown Logger Public, TINYTYPHON, BADNEWS, and an AutoIt [3] backdoor."[28]

---

[22] "Top 25 Threat Actors – 2019 Edition." SBS CyberSecurity. December 12, 2019. https://sbscyber.com/resources/top-25-threat-actors-2019-edition.

[23] "Advanced Persistent Threat Groups." FireEye. Accessed January 16, 2022. https://www.fireeye.com/current-threats/apt-groups.html.

[24] "Top 25 Threat Actors – 2019 Edition." SBS CyberSecurity. December 12, 2019. https://sbscyber.com/resources/top-25-threat-actors-2019-edition.

[25] Ibid.

[26] Settle, Andy, Nicholas Griffin, and Abel Toro. "Monsoon – Analysis of an APT Campaign: Espionage and Data Loss Under the Cover of Current Affairs. Forcepoint Security Labs. Accessed January 16, 2022. https://www.forcepoint.com/sites/default/files/resources/files/forcepoint-security-labs-monsoon-analysis-report.pdf.

[27] Ibid.

[28] Ibid.

1.2.6 Wi-Fi attacks

With the explosive growth in the number of mobile devices over the past decade, wireless (Wi-Fi) networks are now everywhere. Whether you're in an office, hotel, airport, school, or coffee shop, you're likely in range of a Wi-Fi network somewhere.

Of course, as a security professional, your first concern when trying to get connected is "how secure is this Wi-Fi network?" But for the average user, the unfortunate reality is that Wi-Fi connectivity is more about convenience than security.

Thus, the challenge is not only to secure your Wi-Fi networks but also to protect the mobile devices that your organization's employees use to perform work and access potentially sensitive data – no matter where they are or whose network they're on.

Wi-Fi security begins – and ends – with authentication. If you can't control who has access to your wireless network, then you can't protect your network.

1.2.6.1 Wired Equivalent Privacy

The Wired Equivalent Privacy (WEP) protocol was the wireless industry's first attempt at security. As its name falsely implies, WEP was intended to provide data confidentiality equivalent to the security of a wired network. However, WEP had many well-known and well-publicized weaknesses – such as its weak random value, or initialization vector (IV), and key-generation algorithm – and wasn't effective for establishing a secure wireless network.

1.2.6.2 Wi-Fi Protected Access (WPA/WPA2/WPA3)

WPA was published as an interim standard in 2003, quickly followed by WPA2 in 2004. WPA/WPA2 contains improvements to protect against the inherent flaws in WEP. These improvements include changes to the encryption to avoid many of the problems that plagued WEP.

WPA2 can be implemented in different ways. WPA2-Enterprise, also known as WPA2-802.1x mode, uses the *Extensible Authentication Protocol* (EAP) and *Remote Authentication Dial-In User Service* (RADIUS) for authentication. Numerous EAP types are also available for use in WPA2-Enterprise.

However, the use of a *pre-shared key* (PSK) is by far the most common, particularly in homes, small businesses, and guest Wi-Fi networks. WPA2-PSK can be implemented with just the AP

and the client, requiring neither a third-party 802.1*x* authentication server nor individual user accounts.

Key Terms

The *Extensible Authentication Protocol* (EAP) is a widely used authentication framework that includes about 40 different authentication methods.

Remote Authentication Dial-In User Service (RADIUS) is a client-server protocol and software that enables remote access servers to communicate with a central server to authenticate users and authorize access to a system or service.

A *pre-shared key* (PSK) is a shared secret, used in symmetric key cryptography, that has been exchanged between two parties communicating over an encrypted channel.

WPA2-PSK supports 256-bit keys, which require 64 hexadecimal characters. Because requiring users to enter a 64-hexadecimal character key is impractical, WPA2 includes a function that generates a 256-bit key based on a much shorter passphrase created by the administrator of the Wi-Fi network and the *service set identifier* (SSID) of the AP used as a *salt* for the *one-way hash function*.

In WPA2, the name of the SSID is used for the salt. An easy way to make your Wi-Fi security stronger (and make *rainbow table* attacks impractical) is to change your SSID to something that isn't common or easily guessed.

To execute an attack on a WPA2 passphrase, an attacker needs to be able to test a large number of passphrase candidates. So, although WPA2 remains cryptographically secure (the key isn't recoverable by simple observation of the traffic, as with WEP), methods do exist to test passphrases offline by gathering the handshake packets between the AP and a legitimate user.

To collect the necessary packets to crack a WPA2 passphrase, an attacker could passively gather traffic when a legitimate user joins the network. This method requires time, however, because the attacker does not know when someone will join the network.

For an impatient attacker, the solution is to employ an active attack. As long as a legitimate user is already online, the attacker can force the user's client device to disconnect from the AP with forged de-authentication packets. After getting disconnected, the client device will automatically attempt to reconnect, thus providing the attacker with the handshake packets needed for offline passphrase analysis. Thus, unlike with WEP, attacks on WPA2 can be done

Key Terms

A *service set identifier* (SSID) is a case-sensitive, 32-character alphanumeric identifier that uniquely identifies a Wi-Fi network.

In cryptography, a *salt* is randomly generated data that is used as an additional input to a one-way hash function that "hashes" a password or passphrase. The same original text hashed with different salts results in different hash values.

A *one-way hash function* is a mathematical function that creates a unique representation (a hash value) of a larger set of data in a manner that is easy to compute in one direction (input to output) but not in the reverse direction (output to input). The hash function can't recover the original text from the hash value. However, an attacker could attempt to guess what the original text was and see if it produces a matching hash value.

A *rainbow table* is a pre-computed table used to find the original value of a cryptographic hash function.

without spending a significant amount of time in the proximity of the target network after the handshake packets have been captured.

Next, the attacker must recover (or find) the passphrase itself, which requires the following:

- **A test to check millions of potential passphrases until it finds the correct passphrase.** To avoid detection, an attacker can't use the actual target, because the victim would be able to see this attack activity. The alternative is to use an offline method of testing that uses the handshake packets.
- **A methodology to guess passphrases.** The worst-case scenario is to "brute force" the passphrase, trying every possible combination of numbers and characters until a correct value is found. This effort can produce a correct result given enough time and computing power. However, it's much faster to take educated guesses without having to resort to brute force. By using educated guesses on possible passphrase candidates, the attacker can attempt a much shorter list.

This basic process for recovering Wi-Fi passphrases is similar to cracking user passwords. In the early days of password cracking, an attacker might have knowledge of a target system's one-way hash function and a list of the system's user password hash values. However, the attacker had no way to decrypt the password, because the original text isn't recoverable from a hash. But by encrypting a list of words with the same one-way hash function (a dictionary attack), an

attacker can then compare the resulting hash values with the hash values stored for the various user accounts on the system. So, although the password itself isn't decrypted, a given input that produces a given result – a password match – can be found. With the addition of more computing power, an attacker could try longer word lists and a greater number of variations of each word. The process for attacking WPA2 passphrases is similar.

WPA3 was published in 2018 and introduces security enhancements such as more robust brute-force attack protection, improved hot spot and guest access security, simpler integration with devices that have limited or no user interface (such as IoT devices), and a 192-bit security suite. Newer Wi-Fi routers and client devices will likely support both WPA2 and WPA3 to ensure backward compatibility in mixed environments.

According to the Wi-Fi Alliance, WPA3 features include improved security for IoT devices such as smart bulbs, wireless appliances, smart speakers, and other screen-free gadgets that make everyday tasks easier. The Wi-Fi Alliance hasn't outlined the specific details yet, but WPA3 is expected to support a one-touch setup system that will make devices without screens (such as IoT devices and smart speakers like Google Home and Amazon Echo) easier to connect. It will be similar to the existing Wi-Fi Protected Setup protocol, which involves pushing a button on the router to connect a device.

According to a recent *VentureBeat* article, WPA3 also "supports a much stronger encryption algorithm than WPA2 … intended for industrial, defense, and government applications rather than homes and offices. Specifically, it includes a 192-bit security suite that's aligned with the Commercial National Security Algorithm (CNSA) Suite, a feature requested by the Committee on National Security Systems (CNSS), a part of the U.S. National Security Agency [NSA]."[29]

WPA3 provides protection against brute-force dictionary attacks by implementing "a robust handshake [called the Dragonfly protocol, also referred to as Simultaneous Authentication of Equals] that isn't vulnerable to wireless exploits like KRACK, and it hardens security at the time when the network key is exchanged between a device and the access point."[30] By limiting the number of network password attempts on a per-user basis, WPA3 also reduces the efficacy of common dictionary attacks.

"WPA3 introduces Opportunistic Wireless Encryption (OWE), or individualized data encryption, which encrypts every connection between a device and the router with a unique key. Even if

---

[29] Wiggers, Kyle. "What is WPA3, why does it matter, and when can you expect it?" VentureBeat. May 19, 2018. https://venturebeat.com/2018/05/19/what-is-wpa3-why-does-it-matter-and-when-can-you-expect-it/.

[30] Ibid.

the access point doesn't require a password, your device's data won't be exposed to the wider network."[31]

Instead of breaking into a wireless network, an attacker can trick victims into connecting to a wireless network that the attacker controls. These techniques are part of a larger set of attacks known as man-in-the-middle attacks. With a man-in-the-middle exploit in place on a Wi-Fi network, an attacker can serve up practically any content. For example:

- If a user attempts to download a legitimate file, the attacker can send mobile malware instead.
- When a user attempts to visit a legitimate webpage, the attacker can alter the content to exploit a vulnerability that exists in the device's browser, allowing the attacker to further escalate an attack.
- Email addresses and financial account information can be harvested from the connected endpoint, enabling an attacker to create a very targeted and convincing phishing attack to trick even more users on a network into disclosing sensitive information.

1.2.6.3 Evil Twin

Perhaps the easiest way for an attacker to find a victim to exploit is to set up a wireless access point that serves as a bridge to a real network. An attacker can inevitably bait a few victims with "free Wi-Fi access."

The main problem with this approach is that it requires a potential victim to stumble on the access point and connect. The attacker can't easily target a specific victim, because the attack depends on the victim initiating the connection.

A slight variation on this approach is to use a more specific name that mimics a real access point normally found at a particular location – the Evil Twin. For example, if your local airport provides Wi-Fi service and calls it "Airport Wi-Fi," the attacker might create an access point with the same name using an access point that has two radios. Average users cannot easily discern when they are connected to the real access point or a fake one, so this approach would catch a greater number of users than a method that tries to attract victims at random. Still, the user has to select the network, so a bit of chance is involved in trying to reach a particular target.

The main limitation of the Evil Twin attack is that the attacker can't choose the victim. In a crowded location, the attacker will be able to get a large number of people connecting to the

[31] Ibid.

wireless network to unknowingly expose their account names and passwords. However, it's not an effective approach if the goal is to target employees in a specific organization.

1.2.6.4 Jasager

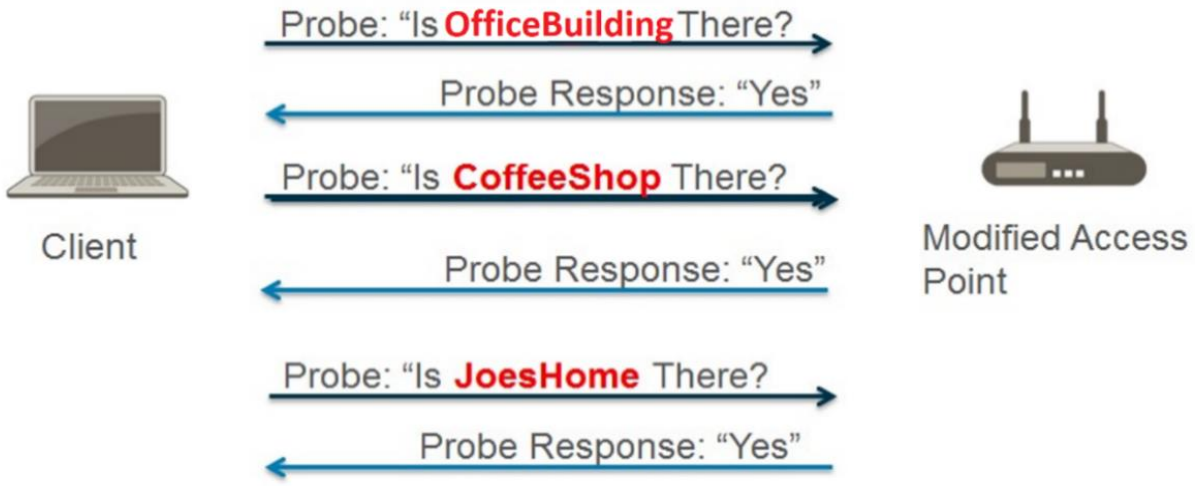
To understand a more targeted approach than the Evil Twin attack, think about what happens when you bring your wireless device back to a location that you've previously visited. For example, when you bring your laptop home, you don't have to choose which access point to use, because your device remembers the details of wireless networks to which it has previously connected. The same goes for visiting the office or your favorite coffee shop.

Your mobile device detects when it is in proximity to a previously known wireless network by sending a beacon out to see if a preferred network is within range. Under normal conditions, when a wireless device sends out a beacon, the non-matching access points ignore it. The beacon goes unanswered, except when it comes within the proximity of the preferred network.

The Jasager attack takes a more active approach toward beacon requests. Jasager (German for "the yes-man") responds to all beacon requests, thus taking a very permissive approach toward who can connect. The user doesn't have to manually choose the attacker's access point. Instead, the attacker pretends to be whatever access point the user normally connects to (see Figure 1-5). Instead of trying to get victims to connect at random, now the attacker simply needs to be within proximity to the target.

Figure 1-5

Jasager pretends to be whichever access point is requested by the client's beacon.



This process intercepts the communication from laptops, mobile phones, and tablets. Many (if not most) 3G/4G/LTE mobile devices automatically switch to Wi-Fi when they recognize that they are near a network that they know.

An attacker can use the same method to capture WPA2 handshake packets (discussed in Section 1.2.6.2) to disconnect users from a Wi-Fi network by using forged de-authentication packets. When the users reconnect, they will unwittingly connect to the modified access point. Unlike the Evil Twin attack, the attacker doesn't have to just wait for a victim to connect to the modified access point; with this approach, everyone who's in the vicinity will automatically connect and become a potential victim.

Jasager runs on any number of devices, but perhaps one of the most effective ways to employ it is with the Pineapple access point. The Pineapple is simply an access point with modified firmware that embeds a number of tools for wireless "penetration" testing. It also has a number of accessories, such as support for cellular USB cards to provide network connectivity when it is otherwise unavailable at the target location, and battery packs to operate as a standalone unit. It's also easily concealed because it can be disguised within any number of housings typically found plugged in at the office.

After the attacker has the victim connected to a malicious access point, the man-in-the-middle attack can proceed, and the attacker not only can observe and capture network traffic but also modify it.

1.2.6.5 SSLstrip

After a user connects to a Wi-Fi network that has been compromised – or to an attacker's Wi-Fi network masquerading as a legitimate network – the attacker can control the content that the victim sees. The attacker simply intercepts the victim's web traffic, redirects the victim's browser to a web server that the attacker controls, and then serves up whatever content the attacker desires.

A man-in-the middle attack can be used to steal a victim's online banking or corporate email account credentials. Normally, this type of traffic would be considered safe because the webpage typically uses Secure Sockets Layer (SSL) encryption. Of course, the average user only knows that a padlock somewhere in the address bar means that their browser is secure, correct?

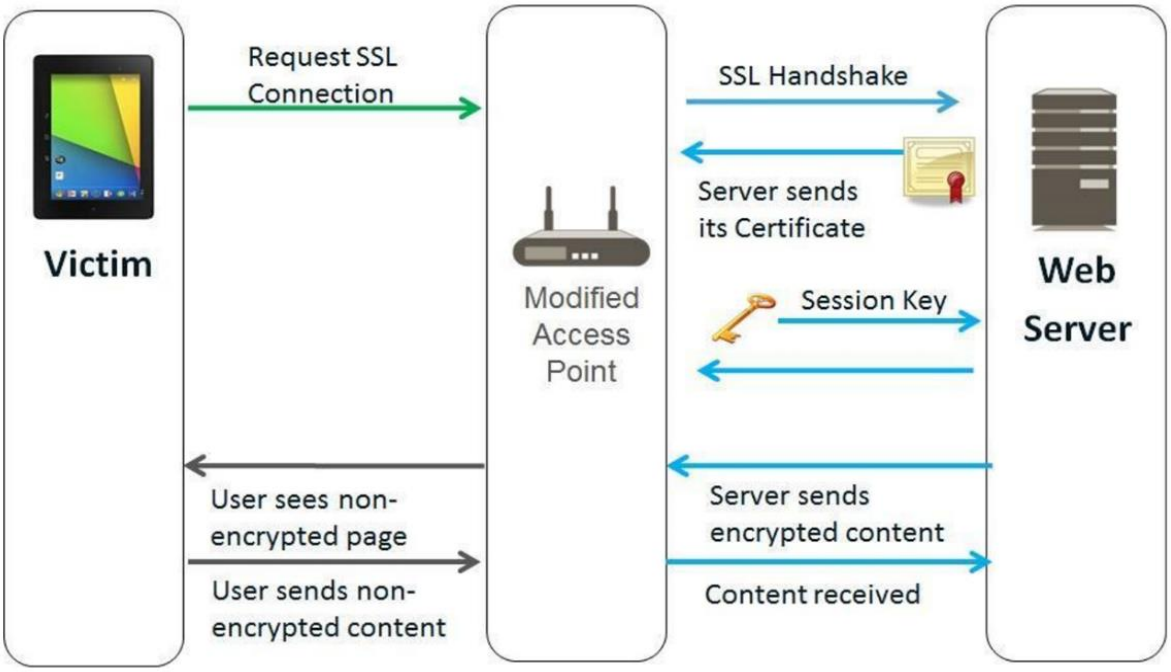
But the padlock appears differently, and in different locations, in different browsers. How does the padlock appear in Internet Explorer? What about Mozilla Firefox, Google Chrome, and Apple Safari? And it appears differently on different smartphones and tablets too. It's no wonder that typical end users – even many security professionals – can be easily tricked.

SSLstrip strips SSL encryption from a "secure" session. When a user connected to a compromised Wi-Fi network attempts to initiate an SSL session, the modified access point intercepts the SSL request (see Figure 1-6). The modified access point then completes the SSL

session on behalf of the victim's device. Then the SSL tunnel between the victim's device and the legitimate secure web server is actually terminated – and decrypted – on the modified access point, thus allowing the attacker to see the victim's credentials, and other sensitive information, in cleartext.

Figure 1-6

Man-in-the-middle with SSLstrip



With SSLstrip, the modified access point displays a fake padlock in the victim's web browser. Webpages can display a small icon called a *favicon* next to a website address in the browser's address bar. SSLstrip replaces the favicon with a padlock that looks like SSL to an unsuspecting

Key Terms

A *favicon* ("favorite icon") is a small file containing one or more small icons associated with a particular website or webpage.

user.

1.2.6.6 Emotet

Emotet is a Trojan, first identified in 2014, that has long been used in spam botnets and ransomware attacks. Recently, it was discovered that a new Emotet variant is using a Wi-Fi

spreader module to scan Wi-Fi networks looking for vulnerable devices to infect.[32] The Wi-Fi spreader module scans nearby Wi-Fi networks on an infected device and then attempts to connect to vulnerable Wi-Fi networks via a brute-force attack. After successfully connecting to a Wi-Fi network, Emotet then scans for non-hidden shares and attempts another brute-force attack to guess usernames and passwords on other devices connected to the network. It then installs its malware payload and establishes C2 communications on newly infected devices.

1.2.6.7 Doppelganger

Doppelganger is an insider attack that targets WPA3-Personal protected Wi-Fi networks. The attacker spoofs the source MAC address of a device that is already connected to the Wi-Fi network and attempts to associate with the same wireless access point.

1.2.6.8 Cookie guzzler

Muted peer and *hasty peer* are variants of the cookie guzzler attack, which exploits the Anti-Clogging Mechanism (ACM) of the Simultaneous Authentication of Equals (SAE) key exchange in WPA3-Personal.

[32] Quinn, James. "Emotet Evolves With New Wi-Fi Spreader." Binary Defense. February 7, 2020. https://www.binarydefense.com/emotet-evolves-with-new-wi-fi-spreader/.

1.2 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Multiple Choice.** Which option describes malicious software or code that typically takes control of, collects information from, or damages an infected endpoint? (Choose one.)

    a) exploit

    b) malware

    c) vulnerability

    d) none of the above

2. **Multiple Choice.** Which option is an important characteristic or capability of advanced malware? (Choose one.)

    a) distributed, fault-tolerant architecture

    b) multifunctionality

    c) hiding techniques such as polymorphism, metamorphism, and obfuscation

    d) all of the above

1.3 Network Security Models

This section describes perimeter-based and Zero Trust network security models.

1.3.1 Perimeter-based network security strategy

Perimeter-based ("castle and moat") network security models date back to the early mainframe era (circa late 1950s), when large mainframe computers were located in physically secure "machine rooms" that could be accessed by only a relatively limited number of remote job entry (RJE) "dumb" terminals that were directly connected to the mainframe and also located in physically secure areas. Today's data centers – including on-premises, private cloud, and public cloud data centers – are the modern equivalent of machine rooms, but perimeter-based physical security is no longer sufficient for several obvious but important reasons:

- Mainframe computers predate the internet. In fact, mainframe computers predate ARPANET, which predates the internet. Today, an attacker uses the internet to remotely gain access, instead of physically breaching the data center perimeter.

- Data centers today are remotely accessed by millions of remote endpoint devices from anywhere and at any time. Unlike the RJEs of the mainframe era, modern endpoints (including mobile devices) are far more powerful than many of the early mainframe computers and are themselves targets.

- The primary value of the mainframe computer was its processing power. The relatively limited data that was produced was typically stored on near-line media, such as tape. Today, data is the target. Data is stored online in data centers and in the cloud, and it is a high-value target for any attacker.

The primary issue with a perimeter-based network security strategy in which countermeasures are deployed at a handful of well-defined ingress and egress points to the network is that the strategy relies on the assumption that everything on the internal network can be trusted. However, this assumption is no longer safe to make, given modern business conditions and computing environments where:

- Remote employees (including WFH and WFA), mobile users, and cloud computing solutions blur the distinction between “internal” and “external”

- Wireless technologies, the proliferation of partner connections, and the need to support guest users introduce countless additional pathways into the network branch offices that may be located in untrusted countries or regions

- Insiders, whether intentionally malicious or just careless, may present a very real security threat

Perimeter-based approach strategies fail to account for:

- The potential for sophisticated cyberthreats to penetrate perimeter defenses, in which case they would then have free passage on the internal network

- Scenarios where malicious users can gain access to the internal network and sensitive resources by using the stolen credentials of trusted users

- The reality that internal networks are rarely homogeneous but instead include pockets of users and resources with inherently different levels of trust or sensitivity that should ideally be separated in any event (for example, research and development and financial systems versus print or file servers)

A broken trust model is not the only issue with perimeter-centric approaches to network security. Another contributing factor is that traditional security devices and technologies (such as port-based firewalls) commonly used to build network perimeters let too much unwanted traffic through. Typical shortcomings in this regard include the inability to:

- Definitively distinguish good applications from bad ones (which leads to overly permissive access control settings)
- Adequately account for encrypted application traffic
- Accurately identify and control users (regardless of where they're located or which devices they're using)
- Filter allowed traffic not only for known application-borne threats but also for unknown ones

The net result is that re-architecting defenses in a way that creates pervasive internal trust boundaries is, by itself, insufficient. You must also ensure that the devices and technologies used to implement these boundaries actually provide the visibility, control, and threat inspection capabilities needed to securely enable essential business applications while still thwarting modern malware, targeted attacks, and the unauthorized exfiltration of sensitive data.

1.3.2 Zero Trust security

Introduced by Forrester Research, the Zero Trust security model addresses some of the limitations of perimeter-based network security strategies by removing the assumption of trust from the equation. With Zero Trust, essential security capabilities are deployed in a way that provides policy enforcement and protection for all users, devices, applications, and data resources, as well as the communications traffic between them, regardless of location.

In particular, with Zero Trust there is no default trust for any entity – including users, devices, applications, and packets – regardless of what it is and its location on or relative to the enterprise network. Verification that authorized entities are always doing only what they're allowed to do also is no longer optional in a Zero Trust model: verification is now mandatory.

These changes imply the following needs:

- The need to establish trust boundaries that effectively compartmentalize the various segments of the internal computing environment. The general idea is to move security functionality closer to the pockets of resources that require protection. In this way,

security can always be enforced regardless of the point of origin of associated communications traffic.

- The need for trust boundaries to do more than just initial authorization and access control enforcement. To "always verify" also requires ongoing monitoring and inspection of associated communications traffic for subversive activities (such as threats).

Benefits of implementing a Zero Trust network include:

- Clearly improved effectiveness in mitigating data loss with visibility and safe enablement of applications, and detection and prevention of cyberthreats
- Greater efficiency for achieving and maintaining compliance with security and privacy mandates, using trust boundaries to segment sensitive applications, systems, and data
- Improved ability to securely enable transformative IT initiatives, such as user mobility, bring your own device (BYOD) and bring your own access (BYOA), infrastructure virtualization, and cloud computing
- Lower total cost of ownership (TCO) with a consolidated and fully integrated security operating platform, rather than a disparate array of siloed, purpose-built security point products

1.3.2.1 Core Zero Trust design principles

The core Zero Trust principles that define the operational objectives of a Zero Trust implementation include:

- **Ensure that all resources are accessed securely, regardless of location.** This principle suggests not only the need for multiple trust boundaries but also increased use of secure access for communication to or from resources, even when sessions are confined to the "internal" network. It also means ensuring that the only devices allowed access to the network have the correct status and settings, have an approved VPN client and proper passcodes, and are not running malware.
- **Adopt a *least privilege* strategy and strictly enforce access control.** The goal is to minimize allowed access to resources as a means to reduce the pathways available for malware and attackers to gain unauthorized access – and to subsequently spread laterally and/or infiltrate sensitive data.

- **Inspect and log all traffic.** This principle reiterates the need to "always verify" while also reinforcing that adequate protection requires more than just strict enforcement of access control. Close and continuous attention must also be given to exactly what "allowed" applications are actually doing, and the only way to accomplish these goals is to inspect the content for threats.

1.3.2.2 Zero Trust conceptual architecture

Traditional security models identify areas where breaches and exploits may occur, the *attack surface*, and you attempt to secure the entire surface. Unfortunately, it is often difficult to identify the entire attack surface. Unauthorized applications, devices, and misconfigured infrastructure can expand that attack surface without your knowledge.

In Zero Trust, you identify a *protect surface* (see Figure 1-7). The protect surface is made up of the network's most critical and valuable data, assets, applications, and services (DAAS). The protect surface is much smaller than the attack surface and should therefore be knowable. Protect surfaces are unique to each organization. Because it contains only what's most critical to an organization's operations, the protect surface is orders of magnitude smaller than the attack surface, and it is always knowable.

With the protect surface identified, you can identify how traffic moves across the organization in relation to the protect surface. Understanding who the users are, which applications they are using, and how they are connecting is the only way to determine and enforce policy that ensures secure access to your data. With an understanding of the interdependencies between the DAAS, infrastructure, services, and users, you should put controls in place as close to the protect surface as possible, creating a micro-perimeter around it. This micro-perimeter moves with the protect surface, wherever it goes.

Key Terms

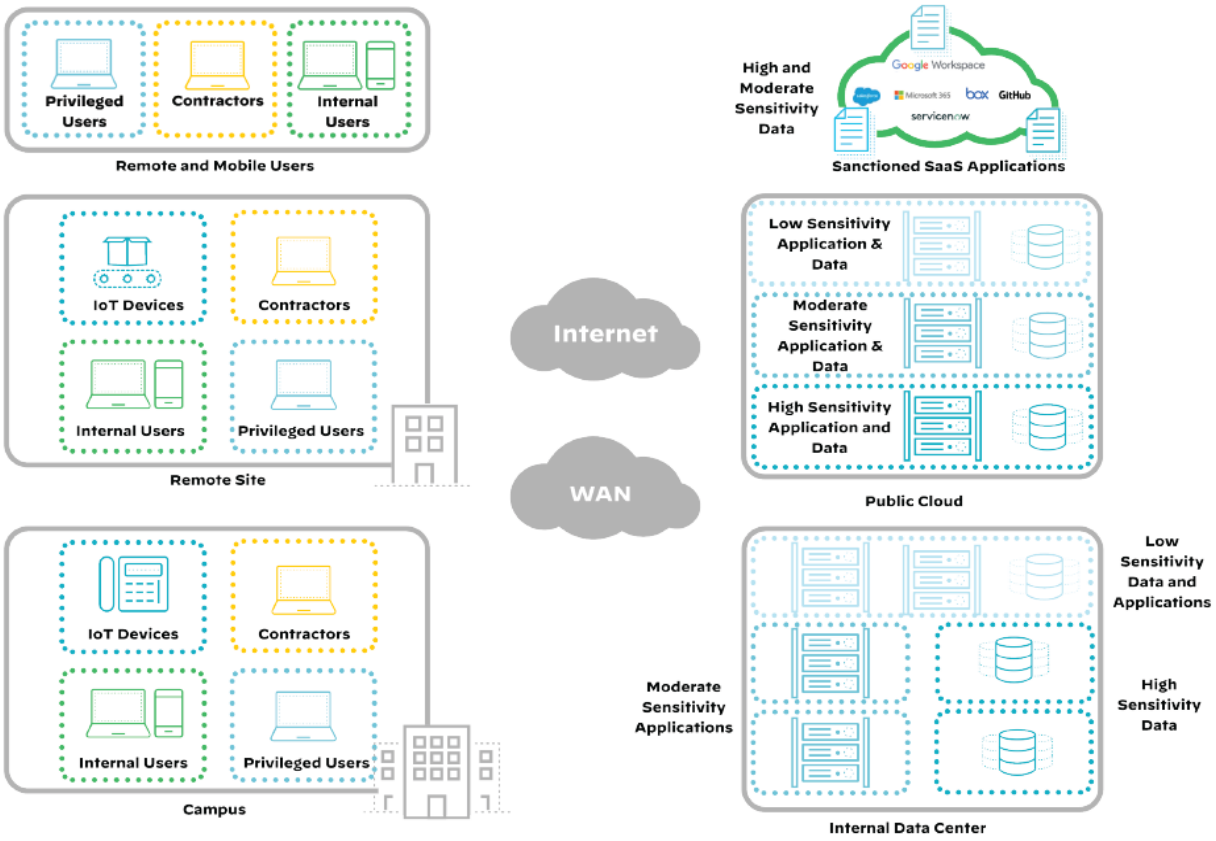
An *attack surface* is any area where breaches and exploits may occur and is comprised of an organization's entire digital footprint.

The principle of *least privilege* in network security requires that only the permission or access rights necessary to perform an authorized task are granted.

A *protect surface* consists of the most critical and valuable *data, assets, applications, and services* (DAAS) on a network.

Figure 1-7

Zero Trust protect surface



In the Zero Trust model, only known and permitted traffic is granted access to the protect surface. A segmentation gateway, typically a next-generation firewall, controls this access. The segmentation gateway provides visibility into the traffic and users attempting to access the protect surface, enforces access control, and provides additional layers of inspection.

Zero Trust policies provide granular control of the protect surface, making sure that users have access to the data and applications they need to perform their tasks but nothing more. This is known as least privilege access.

Additionally, to implement a Zero Trust least privilege access model in the network, the firewall must:

- **Have visibility of and control over the applications and their functionality in the traffic.** Traditional security infrastructure describes applications through ports and protocols. Zero Trust's least privilege access model requires precise control over application use that a port and protocol definition cannot achieve.

- **Be able to allow specific applications and block everything else.** Allowing a specific set of applications through an allow-list and denying everything else significantly reduces the number of ways an organization can be attacked.

- **Dynamically define access to sensitive applications and data based on a user's group membership.** Many traditional security policies define access based on the location of the endpoint in the network. Even if enterprise mobility didn't blur the traditional network boundaries, network location is a poor identifier for a user and their assigned privileges.

- **Dynamically define access from devices or device groups to sensitive applications and data and from users and user groups to specific devices.** This is important in IoT-heavy environments, where devices may access applications and data in the same way a user would. For example, medical equipment may be sending sensitive data to specific applications or repositories. Malicious or even accidental access might disrupt manufacturing equipment or industrial control systems.

- **Be able to validate a user's identity through authentication.** For access to the most sensitive data, the firewall should validate user information obtained from the organization's authentication servers with another authentication method before allowing access. This ensures the traffic is coming from the expected user and not from someone impersonating them.

- **Dynamically define the resources that are associated with the sensitive data or application.** Many data centers and PaaS environments dynamically allocate resources to applications. To ensure that the security posture matches the current resource allocation, the firewall needs to adjust along with the changing environment.

- **Control data by file type and content.** Blocking risky file types reduces the number of ways you can be attacked and reduces the number of ways attackers can exfiltrate data.

The result is granular control that safely allows access to the right applications for the right sets of users while automatically eliminating unwanted, unauthorized, and potentially harmful interactions.

The main components of a Zero Trust conceptual architecture (shown in Figure 1-8) include:

- **Zero Trust Segmentation Platform.** The Zero Trust Segmentation Platform is referred to as a network segmentation gateway by Forrester Research. It is the component used to define internal trust boundaries, meaning that the platform provides the majority of the

security functionality needed to deliver on the Zero Trust operational objectives, including the ability to:

- Enable secure network access
- Granularly control traffic flow to and from resources
- Continuously monitor allowed sessions for any threat activity

Although Figure 1-8 depicts the Zero Trust Segmentation Platform as a single component in a single physical location, in practice – due to performance, scalability, and physical limitations – an effective implementation is more likely to entail multiple instances distributed throughout an organization's network. The solution is also designated as a "platform" to reflect the fact that it is an aggregation of multiple distinct (and potentially distributed) security technologies operating as part of a holistic threat protection framework to reduce the attack surface and correlate information about threats that are found.

- **Trust zones.** Forrester Research refers to a trust zone as a micro core and perimeter (MCAP). A trust zone is a distinct pocket of infrastructure where the member resources not only operate at the same trust level but also share similar functionality. Functionality such as protocols and types of transactions must be shared in order to minimize the number of allowed pathways into and out of a given zone and, in turn, to minimize the potential for malicious insiders and other types of threats to gain unauthorized access to sensitive resources.

  Examples of trust zones shown in Figure 1-8 include the user (or campus) zone, a wireless zone for guest access, a cardholder data zone, database and application zones for multitier services, and a zone for public-facing web applications.

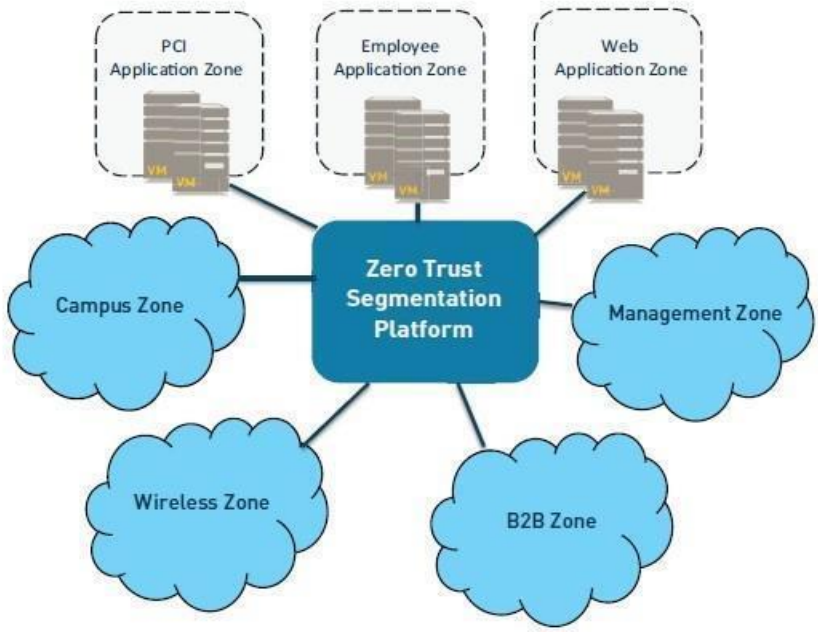
  Remember, too, that a trust zone is not intended to be a "pocket of trust" where systems (and therefore threats) within the zone can communicate freely and directly with each other. For a full Zero Trust implementation, the network would be configured to ensure that *all* communications traffic – including traffic between devices in the same zone – is intermediated by the corresponding Zero Trust Segmentation Platform.

- **Management infrastructure.** Centralized management capabilities are crucial to enabling efficient administration and ongoing monitoring, particularly for implementations involving multiple distributed Zero Trust Segmentation Platforms. A data acquisition network also provides a convenient way to supplement the native monitoring and analysis capabilities for a Zero Trust Segmentation Platform. Session logs

that have been forwarded to a data acquisition network can then be processed by any number of out-of-band analysis tools and technologies intended, for example, to further enhance network visibility, detect unknown threats, or support compliance reporting.

Figure 1-8

Zero Trust conceptual architecture



1.3.2.3 Key Zero Trust criteria and capabilities

The core of any Zero Trust network security architecture is the Zero Trust Segmentation Platform, so you must choose the correct solution. Key criteria and capabilities to consider when selecting a Zero Trust Segmentation Platform include:

- **Secure access.** Consistent secure IPsec and SSL VPN connectivity is provided for all employees, partners, customers, and guests wherever they're located (for example, at remote or branch offices, on the local network, or over the internet). Policies to determine which users and devices can access sensitive applications and data can be defined based on application, user, content, device, device state, and other criteria.

- **Inspection of all traffic.** Application identification accurately identifies and classifies all traffic, regardless of ports and protocols, and evasive tactics, such as port hopping or encryption. Application identification eliminates methods that malware may use to hide from detection and provides complete context into applications, associated content, and threats.

- **Least privileges access control.** The combination of application, user, and content identification delivers a positive control model that allows organizations to control interactions with resources based on an extensive range of business-relevant attributes, including the specific application and individual functions being used, user and group identity, and the specific types or pieces of data being accessed (such as credit card or Social Security numbers). This results in truly granular access control that safely enables the correct applications for the correct sets of users while automatically preventing unwanted, unauthorized, and potentially harmful traffic from gaining access to the network.

- **Cyberthreat protection.** A combination of anti-malware, intrusion prevention, and cyberthreat prevention technologies provides comprehensive protection against both known and unknown threats, including threats on mobile devices. Support for a closed-loop and highly integrated defense also ensures that inline enforcement devices and other components in the threat protection framework are automatically updated.

- **Coverage for all security domains.** Virtual and hardware appliances establish consistent and cost-effective trust boundaries throughout an organization's entire network, including in remote or branch offices, for mobile users, at the internet perimeter, in the cloud, at ingress points throughout the data center, and for individual areas wherever they might exist.

1.3.2.4 Implementing a Zero Trust design

Implementation of a Zero Trust network security model doesn't require a major overhaul of an organization's network and security infrastructure. A Zero Trust design architecture can be implemented in a way that requires only incremental modifications to the existing network and is completely transparent to its users. Advantages of such a flexible, non-disruptive deployment approach include minimizing the potential impact on operations and being able to spread the required investment and work effort over time.

To get started, you can configure a Zero Trust Segmentation Platform in listen-only mode to obtain a detailed picture of traffic flows throughout the network, including where, when, and to what extent specific users are using specific applications and data resources.

With a detailed understanding of the network traffic flows in the environment, the next step is to define trust zones and incrementally establish corresponding trust boundaries based on relative risk and/or sensitivity of the data involved:

- Deploy devices in appropriate locations to establish internal trust boundaries for defined trust zones.

- Configure the appropriate enforcement and inspection policies to effectively put each trust boundary “online.”

Next, you can progressively establish trust zones and boundaries for other segments of the computing environment based on their relative degree of risk. Examples of where secure trust zones can be established are:

- IT management systems and networks (where a successful breach could lead to the entire network becoming compromised)
- Partner resources and connections (business to business, or B2B)
- High-profile, customer-facing resources and connections (business to consumer, or B2C)
- Branch offices in risky countries or regions, followed by all other branch offices
- Guest access networks (both wireless and wired)
- Campus networks

Zero Trust principles and concepts must be implemented at major access points to the internet. You will have to replace or augment legacy network security devices with a Zero Trust Segmentation Platform at this deployment stage to gain all of the requisite capabilities and benefits of a Zero Trust security model.

1.3 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Short Answer.** What is the primary issue with a perimeter-based network security strategy today?
2. **Multiple Choice.** A Zero Trust network security model is based on which security principle? (Choose one.)

    a) due diligence

    b) least privilege

    c) non-repudiation

    d) negative control

1.4 Security Operating Platform

Cybercrime and the types of security threats continue to evolve, challenging organizations to keep up as network boundaries and attack surfaces expand. Security breaches and intellectual property loss can have a huge impact on organizations. Current approaches to security, which focus mainly on detection and remediation, are inadequate to sufficiently address the rise in volume and sophistication of attacks. Cybercriminals leverage automation and big data analytics to execute massively scalable and increasingly effective attacks against their targets. They often share data and techniques with other threat actors to keep their approach ahead of point security products. Cybercriminals are not the only threat: Employees may often unknowingly violate corporate compliance and expose critical data in locations such as the public cloud.

With the rapid evolution of applications moving to the cloud, decentralization of IT infrastructure, and the increased threat landscape, the result has been a loss of visibility and control for organizations. Devices are proliferating and the network perimeter has all but disappeared, leaving enterprise security teams struggling to safely enable and protect their businesses, customers, and users. With new threats growing in number and sophistication, organizations are finding that traditional security products and approaches are less and less capable of protecting their networks against today's advanced cyberattacks.

At the same time, application development and IT operations teams are accelerating the delivery of new applications to drive business growth by adopting DevOps tools and methodologies, cloud and container technologies, big data analytics, and automation and orchestration. Meanwhile, applications are increasingly accessible. The result is an incredibly complex network that introduces significant business risk. Organizations must minimize this risk without slowing down the business.

Therefore, a different approach to security is needed. Defenders must replace siloed point products with security innovations that are tightly integrated. Security requires simplicity. The Palo Alto Networks Security Operating Platform consists of a tightly integrated system of components and services, including a partner ecosystem, that delivers consistent security across the network, endpoints, and cloud. The Security Operating Platform is a fully integrated system that simplifies security by leveraging consolidated threat intelligence information, automation, machine learning, and data analytics (see Figure 1-9).

Figure 1-9

Palo Alto Networks Security Operating Platform

The Security Operating Platform is designed so that security teams can operate simply and efficiently to protect their organizations. The platform prevents successful attacks and stops attacks in progress while providing consistent protection to secure the enterprise, the cloud, and an organization's future. Rooted in prevention, the Security Operating Platform is designed and purpose-built to counter attacks before they can breach an organization's environment.

The Security Operating Platform's prevention architecture allows organizations to reduce threat exposure by first enabling applications for all users or devices in any location and then by preventing threats within application flows. This is achieved by tying application use to user identities across physical, cloud-based, and software-as-a-service (SaaS) environments.

To enable the prevention of successful cyberattacks, the Security Operating Platform delivers four key capabilities:

1. **Provide full visibility.** To understand the full context of an attack, visibility of all users and devices is provided across the organization's network, endpoint, cloud, and SaaS applications.

2. **Reduce the attack surface.** Best-of-breed technologies that are natively integrated provide a prevention architecture that inherently reduces the attack surface. This type of architecture allows organizations to exert positive control based on applications, users, and content, with support for open communication, orchestration, and visibility.

3. **Prevent all known threats, fast.** A coordinated security platform accounts for the full scope of an attack, across the various security controls that compose the security posture, enabling organizations to quickly identify and block known threats.

4. **Detect and prevent new, unknown threats with automation.** Building security that simply detects threats and requires a manual response is too little, too late. Automated creation and delivery of near-real-time protections against new threats to the various security solutions in the organization's environments enable dynamic policy updates. These updates are designed to allow enterprises to scale defenses with technology, rather than people.

Security should not be a barrier to the adoption of new mobility, SaaS, public, or private cloud technologies that enable productivity. With a natively integrated, prevention-first security platform in place, organizations can securely adopt innovative, productivity-enhancing applications and technologies, all while maintaining a comprehensive and consistent prevention-oriented enterprise security posture.

1.4 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Fill in the Blank.** _____ is a tightly integrated system of components and services, including a partner ecosystem, that delivers consistent security across the network, endpoints, and cloud.

2. **Multiple Choice.** Which three options are key components of the Security Operating Platform? (Choose three.)

    a) network security

    b) advanced endpoint protection

    c) cloud security

    d) application development security

Conclusion

Palo Alto Networks is helping to address the world's greatest security challenges with continuous innovation that seizes the latest breakthroughs in artificial intelligence, analytics, automation, and orchestration. By delivering an integrated platform and empowering a growing ecosystem of partners, Palo Alto Networks is at the forefront of protecting tens of thousands of organizations across clouds, networks, and mobile devices.

The broad portfolio of Palo Alto Networks security technologies and solutions addresses three essential areas of cybersecurity strategy:

- **Secure the Enterprise (Strata):**

    - **Palo Alto Networks PA-Series, VM-Series, and K2-Series next-generation firewalls** (discussed in Section 2.6.1) are the cornerstone of enterprise network security. Powered by PAN-OS® software, these next-generation firewalls leverage App-ID, User-ID, and Content-ID to provide complete visibility and control of the applications in use across all users, devices, and locations.

    - **Cloud-based subscription services** (discussed in Section 2.6.2), including DNS Security, URL Filtering, Threat Prevention, WildFire® malware prevention and others, deliver real-time advanced predictive analytics, AI and machine learning,

exploit/malware/C2 threat protection, and global threat intelligence to the Palo Alto Networks Security Operating Platform.

- **Panorama** (discussed in Section 2.6.3) network security management enables centralized control, log collection, and policy workflow automation across all next-generation firewalls (scalable to tens of thousands of firewalls) from a single pane of glass.

- **Secure the Cloud (Prisma):**

  - **Prisma Cloud** (discussed in Section 3.4) is the industry's most comprehensive threat protection, governance, and compliance offering. It dynamically discovers cloud resources and sensitive data across AWS, GCP, and Azure to detect risky configurations and identify network threats, suspicious user behavior, malware, data leakage, and host vulnerabilities. It eliminates blind spots across cloud environments and provides continuous protection with a combination of rule-based security policies and class-leading machine learning.

  - **Prisma Access (SASE)** (discussed in Section 3.5.2) helps organizations deliver consistent security to all remote networks and mobile users. It is a generational step forward in cloud security, using a cloud-delivered architecture to connect all users to all applications. All of your users, whether at your headquarters, in branch offices, or on the road, connect to Prisma Access to safely use cloud and data center applications, as well as the internet. Prisma Access consistently inspects all traffic across all ports and provides bidirectional software-defined wide-area networking (SD-WAN) to enable branch-to-branch and branch-to-headquarters traffic.

  - **Prisma SaaS** (discussed in Section 3.5.3) functions as a multimode cloud access security broker (CASB), offering inline and API-based protection working together to minimize the range of cloud risks that can lead to breaches. With a fully cloud-delivered approach to CASB, you can secure your SaaS applications through the use of inline protections to safeguard inline traffic with deep application visibility, segmentation, secure access, and threat prevention, as well as API-based protections to connect directly to SaaS applications for data classification, data loss prevention, and threat detection.

- **Secure the Future (Cortex):**

  - **Cortex XDR** (discussed in Section 4.4.1) breaks the silos of traditional detection and response by natively integrating network, endpoint, and cloud data to stop

sophisticated attacks. By taking advantage of machine learning and AI models across all data sources, Cortex XDR identifies unknown and highly evasive threats from managed and unmanaged devices.

- **Cortex XSOAR** (discussed in Section 4.4.2) is the only security orchestration, automation, and response (SOAR) platform that combines security orchestration, incident management, and interactive investigation to serve security teams across the incident lifecycle.

- **Cortex XSOAR Threat Intelligence Management (TIM)** (discussed in Section 4.4.3) is a threat intelligence platform that automatically maps threat information to incidents happening in your network to quickly identify the connections between threat actors and attack techniques previously unknown in your environment.

- **Cortex Data Lake** (discussed in Section 4.4.4) enables AI-based innovations for cybersecurity with the industry's only approach to normalizing your enterprise's data. It automatically collects, integrates, and normalizes data across your security infrastructure. The cloud-based service is ready to scale from the start, eliminating the need for local compute or storage, providing assurance in the security and privacy of your data.

Module 2 – Fundamentals of Network Security

Knowledge Objectives

- Describe the basic operation of computer networks and the internet, including common networking devices, routed and routing protocols, different types of area networks and topologies, the Domain Name System (DNS), and the internet of things (IoT).
- Explain the functions of physical, logical, and virtual addressing in networking.
- Discuss IPv4 and IPv6 addressing and subnetting fundamentals.
- Discuss the OSI reference model and the TCP/IP model, including packet analysis, protocol and packet filtering, and TCP/IP encapsulation.
- Describe network security technologies, including firewalls, intrusion detection systems (IDSs) and intrusion prevention systems (IPSs), web content filters, virtual private networks (VPNs), data loss prevention (DLP), and unified threat management (UTM).
- Discuss endpoint security challenges and solutions, including malware protection, anti-spyware software, personal firewalls, host-based intrusion prevention systems (HIPSs), and mobile device management (MDM).
- Discuss network operations concepts, including server and system administration, directory services, and structured host and network troubleshooting.
- Discuss the fundamental components of a next-generation firewall and explain the basic operation of a next-generation firewall. Compare and contrast traditional port-based firewalls and next-generation firewalls.

2.0 The Connected Globe

With almost 5 billion internet users worldwide in 2022, which represents well over half the world's population, the internet connects businesses, governments, and people across the globe. Our reliance on the internet will continue to grow, with nearly 30 billion devices and "things" – including autonomous vehicles, household appliances, wearable technology, and

more – connecting to the internet of things (IoT) and nearly 9 billion worldwide smartphone subscriptions that will use a total of 160 exabytes (EB) of monthly data by 2025.[33]

2.0.1 The NET: How things connect

In the 1960s, the U.S. Defense Advanced Research Project Agency (DARPA) created ARPANET, the precursor to the modern internet. ARPANET was the first packet-switched network. A packet-switched network breaks data into small blocks (packets), transmits each individual packet from node to node toward its destination, and then reassembles the individual packets in the correct order at the destination.

Today, hundreds of millions of routers deliver Transmission Control Protocol/Internet Protocol (TCP/IP) packets using various routing protocols (discussed in Section 2.2.3) across local-area networks and wide-area networks. The Domain Name System (DNS, discussed in Section 2.0.4) enables internet addresses, such as www.paloaltonetworks.com, to be translated into routable IP addresses.

2.0.2 Introduction to networking devices

Routers are physical or virtual devices that send data packets to destination networks along a network path using logical addresses (discussed in Section 2.1). Routers use various routing protocols to determine the best path to a destination, based on variables such as bandwidth, cost, delay, and distance. A wireless router combines the functionality of a router and a wireless access point (AP) to provide routing between a wired and wireless network. An AP is a network device that connects to a router or wired network and transmits a Wi-Fi signal so that wireless devices can connect to a wireless (or Wi-Fi) network. A *wireless repeater* rebroadcasts the wireless signal from a wireless router or AP to extend the range of a Wi-Fi network.

A *hub* (or *concentrator*) is a network device that connects multiple devices – such as desktop computers, laptop docking stations, and printers – on a local-area network (LAN). Network traffic that is sent to a hub is broadcast out of all ports on the hub, which can create network congestion and introduces potential security risks (broadcast data can be intercepted).

A *switch* is essentially an intelligent hub that uses physical addresses (discussed in Section 2.1) to forward data packets to devices on a network. Unlike a hub, a switch is designed to forward data packets only to the port that corresponds to the destination device. This transmission

[33] "Ericsson Mobility Report, November 2019." Ericsson. November 2019. https://www.ericsson.com/en/mobility-report.

method (referred to as micro-segmentation) creates separate network segments and effectively increases the data transmission rates available on the individual network segments. Also, a switch can be used to implement *virtual LANs* (VLANs), which logically segregate a network and limit *broadcast domains* and *collision domains*.

Key Terms

A *router* is a network device that sends data packets to a destination network along a network path.

A *wireless repeater* rebroadcasts the wireless signal from a wireless router or AP to extend the range of a Wi-Fi network.

A *hub* (or *concentrator*) is a device used to connect multiple networked devices on a local-area network (LAN).

A *switch* is an intelligent hub that forwards data packets only to the port associated with the destination device on a network.

A *virtual LAN* (VLAN) is a logical network that is created within a physical LAN.

A *broadcast domain* is the portion of a network that receives broadcast packets sent from a node in the domain.

A *collision domain* is a network segment on which data packets may collide with each other during transmission.

2.0.3 Area networks and topologies

Most computer networks are broadly classified as either local-area networks (LANs) or wide-area networks (WANs).

A *local-area network* (LAN) is a computer network that connects end-user devices such as laptop and desktop computers, servers, printers, and other devices so that applications, databases, files, file storage, and other networked resources can be shared among authorized users on the LAN. A LAN operates across a relatively small geographic area (such as a floor, a building, or a group of buildings), typically at speeds of up to 10 megabits per second (Mbps – Ethernet), 100Mbps (Fast Ethernet), 1,000Mbps (or 1 gigabit per second [1Gbps] – Gigabit Ethernet) on wired networks and 11Mbps (802.11b), 54Mbps (802.11a and g), 450Mbps (802.11n), 1.3Gbps (802.11ac), and 14Gbps (802.11ax – theoretical) on wireless networks. A LAN can be wired, wireless, or a combination of wired and wireless. Networking equipment commonly used in LANs include *bridges*, *repeaters*, switches, and wireless access points (APs).

Key Terms

A *local-area network* (LAN) connects computers, servers, printers, and other devices so that applications, databases, files and file storage, and other networked resources can be shared across a relatively small geographic area, such as a floor, a building, or a group of buildings.

A *bridge* is a wired or wireless network device that extends a network or joins separate network segments.

A *repeater* is a network device that boosts or retransmits a signal to physically extend the range of a wired or wireless network.

Two basic network topologies (with many variations) are commonly used in LANs:

- **Star.** Each node on the network is directly connected to a switch, hub, or concentrator, and all data communications must pass through the switch, hub, or concentrator. The switch, hub, or concentrator can thus become a performance bottleneck or single point of failure in the network. A star topology is ideal for practically any size environment and is the most commonly used basic LAN topology.
- **Mesh.** All nodes are interconnected to provide multiple paths to all other resources. A mesh topology may be used throughout the network or only for the most critical network components, such as routers, switches, and servers, to eliminate performance bottlenecks and single points of failure.

Key Terms

In a *ring topology*, all nodes are connected in a closed loop that forms a continuous ring and all communication travels in a single direction around the ring. Ring topologies were common in token ring networks.

In a *bus (or linear bus) topology*, all nodes are connected to a single cable (the backbone) that is terminated on both ends. In the past, bus networks were commonly used for very small networks because they were inexpensive and relatively easy to install.

A *wide-area network* (WAN) is a computer network that connects multiple LANs or other WANs across a relatively large geographic area, such as a small city, a region or country, a global enterprise network, or the entire planet (for example, the internet).

A WAN connects networks using telecommunications circuits and technologies such as *multiprotocol label switching* (MPLS), *broadband cable*, *digital subscriber line* (DSL), *fiber optic, optical carrier* (for example, OC-3), and *T-carrier* (for example, T-1) at various speeds typically ranging from 256Kbps to several hundred megabits per second. Examples of networking equipment commonly used in WANs include access servers, channel service units (CSUs) and data service units (DSUs), firewalls, modems, routers, virtual private network (VPN) gateways, and WAN switches.

Key Terms

A *wide-area network* (WAN) is a computer network that connects multiple LANs or other WANs across a relatively large geographic area, such as a small city, a region or country, a global enterprise network, or the entire planet (for example, the internet).

Multiprotocol label switching (MPLS) is a networking technology that routes traffic using the shortest path based on "labels," rather than network addresses, to handle forwarding over private wide-area networks.

Broadband cable is a type of high-speed internet access that delivers different upload and download data speeds over a shared network medium. The overall speed varies depending on the network traffic load from all the subscribers on the network segment.

Digital subscriber line (DSL) is a type of high-speed internet access that delivers different upload and download data speeds. The overall speed depends on the distance from the home or business location to the provider's central office (CO).

Fiber optic technology converts electrical data signals to light and delivers constant data speeds in the upload and download directions over a dedicated fiber optic cable medium. Fiber optic technology is much faster and more secure than other types of network technology.

Optical carrier is a specification for the transmission bandwidth of digital signals on synchronous optical networking (SONET) fiber optic networks. Optical carrier transmission rates are designated by the integer value of the multiple of the base rate (51.84Mbps). For example, OC-3 designates a 155.52Mbps (3 x 51.84) network and OC-192 designates a 9953.28Mbps (192 x 51.84) network.

Traditional WANs rely on physical routers to connect remote or branch users to applications hosted on data centers. Each router has a data plane, which holds the information, and a control plane, which tells the data where to go. Where data flows is typically determined by a

network engineer or administrator who writes rules and policies, often manually, for each router on the network – a process that can be time-consuming and prone to human error.

Key Terms

T-carrier is a full-duplex digital transmission system that uses multiple pairs of copper wire to transmit electrical signals over a network. For example, a T-1 circuit consists of two pairs of copper wire – one pair transmits, the other pair receives – that are multiplexed to provide a total of 24 channels, each delivering 64Kbps of data, for a total bandwidth of 1.544Mbps.

A *software-defined wide-area network* (SD-WAN) separates the control and management processes from the underlying networking hardware, making them available as software that can be easily configured and deployed. A centralized control pane means network administrators can write new rules and policies, then configure and deploy them across an entire network at once.

SD-WAN makes it easier to manage and direct traffic across a network. With traditional networking approaches like MPLS, traffic created in the branch is returned, or "backhauled," to a centralized internet security point in a headquarters' data center. Backhauling traffic can lower application performance, which leads to reduced productivity and poor user experience. Because MPLS networks are private networks built for one given organization, they are considered reliable and secure, but they are expensive. Moreover, MPLS is not designed to handle the high volumes of WAN traffic that result from software-as-a-service (SaaS) applications and cloud adoption.

Compared to traditional WANs, SD-WANs can manage multiple types of connections, including MPLS, broadband, *Long-Term Evolution* (LTE) and others, as well as support applications hosted in data centers, public and private clouds, and SaaS services. SD-WAN can route application traffic over the best path in real time. In the case of cloud, SD-WAN can forward internet- and cloud-bound traffic directly from the branch without backhauling.

Key Terms

A *software-defined wide-area network* (SD-WAN) separates the network control and management processes from the underlying hardware in a wide-area network and makes them available as software.

Long-Term Evolution (LTE) is a type of 4G cellular connection that provides fast connectivity primarily for mobile internet use.

SD-WAN offers many benefits to geographically distributed organizations, including:

- **Simplicity.** Because each device is centrally managed with routing based on application policies, WAN managers can create and update security rules in real time as network requirements change. In addition, by combining SD-WAN with zero-touch provisioning – a feature that helps automate the deployment and configuration processes – organizations can further reduce the complexity, resources, and operating expenses required to turn up new sites.

- **Improved performance.** By allowing efficient access to cloud-based resources without the need to backhaul traffic to centralized locations, organizations can provide a better user experience.

- **Reduced costs.** Network administrators can supplement or substitute expensive MPLS with broadband and other connectivity options.

The hierarchical internetworking model is a best-practice network design that was originally proposed by Cisco and is comprised of three layers:

- **Access.** User endpoints and servers connect to the network at this layer, typically via network switches. Switches at this layer may perform some Layer 3 (discussed in Section 2.2.1) functions and may also provide electrical power via *Power over Ethernet* (PoE) ports to other equipment connected to the network, such as wireless APs or VoIP phones.

- **Distribution.** This layer performs any compute-intensive routing and switching functions on the network, such as complex routing, filtering, and *quality of service* (QoS). Switches at this layer may be Layer 7 (discussed in Section 2.2.1) switches and connect to lower-end Access layer switches and higher-end Core layer switches.

- **Core.** This layer is responsible for high-speed routing and switching. Routers and switches at this layer are designed for high-speed packet routing and forwarding.

Key Terms

Power over Ethernet (PoE) is a network standard that provides electrical power to certain network devices over Ethernet cables.

Quality of service (QoS) is the overall performance of specific applications or services on a network including error rate, bit rate, throughput, transmission delay, availability, and jitter. QoS policies can be configured on certain network and security devices to prioritize certain traffic (such as voice or video) over other, less performance-intensive traffic.

In addition to LANs and WANs, many other types of area networks are used for different purposes:

- Campus area networks (CANs) and wireless campus area networks (WCANs) connect multiple buildings in a high-speed network (for example, across a corporate or university campus).
- Metropolitan area networks (MANs) and wireless metropolitan area networks (WMANs) extend networks across a relatively large area, such as a city.
- Personal area networks (PANs) and wireless personal area networks (WPANs) connect an individual's electronic devices – such as laptop computers, smartphones, tablets, virtual personal assistants (for example, Amazon Alexa, Apple Siri, Google Assistant, and Microsoft Cortana), and wearable technology – to each other or to a larger network.
- Storage area networks (SANs) connect servers to a separate physical storage device (typically a disk array).
- Value-added networks (VANs) are a type of extranet that allows businesses within an industry to share information or integrate shared business processes.
- Virtual local-area networks (VLANs) segment broadcast domains in a LAN, typically into logical groups (such as business departments). VLANs are created on network switches.
- Wireless local-area networks (WLANs), also known as Wi-Fi networks, use wireless access points (APs) to connect wireless-enabled devices to a wired LAN.
- Wireless wide-area networks (WWANs) extend wireless network coverage over a large area, such as a region or country, typically using mobile cellular technology.

2.0.4 Domain Name System

The *Domain Name System* (DNS) is a distributed, hierarchical internet database that maps *fully qualified domain names* (FQDNs) for computers, services, and other resources – such as a website address (also known as a uniform resource locator, or URL) – to IP addresses (discussed in Sections 2.1 and 2.1.1), similar to how a contact list on a smartphone maps the names of businesses and individuals to phone numbers. To create a new domain name that will be accessible via the internet, you must register your unique domain name with a *domain name registrar*, such as GoDaddy or Network Solutions. This registration is similar to listing a new phone number in a phone directory. DNS is critical to the operation of the internet.

A root name server is the *authoritative* name server for a DNS root zone. Worldwide, 13 root name servers (actually, 13 networks comprising hundreds of root name servers) are configured. They are named a.root-servers.net through m.root-servers.net. DNS servers are typically configured with a root hints file that contains the names and IP addresses of the root servers.

Key Terms

The *Domain Name System* (DNS) is a hierarchical distributed database that maps the fully qualified domain name (FQDN) for computers, services, or any resource connected to the internet or a private network to an IP address.

A *fully qualified domain name* (FQDN) is the complete domain name for a specific computer, service, or resource connected to the internet or a private network.

A *domain name registrar* is an organization that is accredited by a *top-level domain* (TLD) registry to manage domain name registrations.

A *top-level domain* (TLD) is the highest-level domain in DNS, represented by the last part of an FQDN (for example, .com and .edu). The most commonly used TLDs are generic top-level domains (gTLDs) (such as .com, edu, .net, and .org) and country-code top-level domains (ccTLDs) (such as .ca and .us).

An *authoritative* DNS server is the system of record for a given domain.

A host (such as a web browser on a desktop computer) on a network that needs to connect to another host (such as a web server on the internet) must first translate the name of the destination host from its URL to an IP address. The connecting host (the DNS client) sends a DNS request to the IP address of the DNS server that is specified in the network configuration of the DNS client. If the DNS server is authoritative for the destination domain, the DNS server resolves the IP address of the destination host and answers the DNS request from the DNS

client. For example, consider attempting to connect to an *intranet* server on your internal network from the desktop computer in your office. If the DNS server address that is configured on your computer is an internal DNS server that is authoritative for your intranet domain, the DNS server resolves the IP address of the intranet server. Your computer then encapsulates the resolved destination IP address in the *Hypertext Transfer Protocol* (HTTP) or *Hypertext Transfer Protocol Secure* (HTTPS) request packets that are sent to the intranet server.

Key Terms

An *intranet* is a private network that provides information and resources – such as a company directory, human resources policies and forms, department or team files, and other internal information – to an organization's users. Like the internet, an intranet uses the HTTP and/or HTTPS protocols, but access to an intranet is typically restricted to an organization's internal users. Microsoft SharePoint is a popular example of intranet software.

Hypertext Transfer Protocol (HTTP) is an application protocol used to transfer data between web servers and web browsers.

Hypertext Transfer Protocol Secure (HTTPS) is a secure version of HTTP that uses Secure Sockets Layer (SSL) or Transport Layer Security (TLS) encryption.

If a DNS server is not authoritative for the destination domain (for example, an internet website address), then the DNS server performs a *recursive* query (if it is in fact configured to perform recursive queries) to obtain the IP address of the authoritative DNS server. The non-authoritative DNS server then sends the original DNS request to the authoritative DNS server. This is a top-down process in which the DNS server first consults its root hints file and queries a root name server to identify the authoritative DNS server for the top-level domain, or TLD (for example, .com), associated with the DNS query. The DNS server then queries the TLD server to identify the authoritative server for the specific domain that is being queried (for example, paloaltonetworks.com). This process continues until the authoritative server for the FQDN is identified and queried. The recursive DNS server then answers the original DNS client's request with the DNS information from the authoritative DNS server.

DNS over HTTPS (DoH) is a more secure implementation of the DNS protocol that uses HTTPS to encrypt data between the DNS client and the DNS resolver.

Key Terms

A *recursive* DNS query is performed (if the DNS server allows recursive queries) when a DNS server is not authoritative for a destination domain. The non-authoritative DNS server obtains the IP address of the authoritative DNS server for the destination domain and sends the original DNS request to that server to be resolved.

DNS over HTTPS (DOH) uses the HTTPS protocol to encrypt DNS traffic.

The basic DNS record types are as follows:

- **A (IPv4)** or **AAAA (IPv6)** (Address). Maps a domain or subdomain to an IP address or to multiple IP addresses.
- **CNAME** (Canonical Name). Maps a domain or subdomain to another hostname.
- **MX** (Mail Exchanger). Specifies the hostname or hostnames of email servers for a domain.
- **PTR** (Pointer). Points to a CNAME. Commonly used for reverse DNS lookups that map an IP address to a host in a domain or subdomain.
- **SOA** (Start of Authority). Specifies authoritative information about a DNS zone such as primary name server, email address of the domain administrator, and domain serial number.
- **NS** (Name Server). The NS record specifies an authoritative name server for a given host.
- **TXT** (Text). Stores text-based information.

2.0.5 The internet of things

In 2022, there were more than 11.5 billion internet of things (IoT) devices worldwide,[34] including *machine-to-machine* (M2M), wide-area IoT, short-range IoT, massive-and-critical IoT, and *multi-access edge computing* (MEC) devices.

[34] Vailshery, Lionel Sujay. "Number of IoT connected devices worldwide 2019-2030." March 17, 2022. https://statista.com/statistics/1183457/iot-connected-devices-worldwide.

Key Terms

Machine-to-machine (M2M) devices are networked devices that exchange data and can perform actions without manual human interaction.

Multi-access edge computing (MEC) is defined by the European Telecommunications Standards Institute (ETSI) as an environment "characterized by ultra-low latency and high bandwidth as well as real-time access to radio network information that can be leveraged by applications."

IoT connectivity technologies are broadly categorized as follows:

- **Cellular:**
    - **2G/2.5G.** Due to the low cost of 2G modules, relatively long battery life, and large installed base of 2G sensors and M2M applications, 2G connectivity remains a prevalent and viable IoT connectivity option.
    - **3G.** IoT devices with 3G modules use either Wideband Code Division Multiple Access (W-CDMA) or Evolved High Speed Packet Access (HSPA+ and Advanced HSPA+) to achieve data transfer rates of between 384Kbps and 168Mbps.
    - **4G/Long-Term Evolution (LTE).** 4G/LTE networks enable real-time IoT use cases, such as autonomous vehicles, with 4G LTE Advanced Pro delivering speeds in excess of 3Gbps and less than 2 milliseconds of latency.
    - **5G.** 5G cellular technology provides significant enhancements compared to 4G/LTE networks and is backed by ultra-low latency, massive connectivity and scalability for IoT devices, more efficient use of licensed spectrum, and network slicing for application traffic prioritization.
- **Satellite:**
    - **C-band.** C-band satellite operates in the 4 to 8 gigahertz (GHz) range. It is used in some Wi-Fi devices and cordless phones, as well as surveillance and weather radar systems.
    - **L-band.** L-band satellite operates in the 1 to 2GHz range. It is commonly used for radars, global positioning systems (GPSs), radio, and telecommunications applications.

- **Short-range wireless:**

    - **Adaptive Network Technology + (ANT+).** ANT+ is a proprietary multicast wireless sensor network technology primarily used in personal wearables, such as sports and fitness sensors.

    - **Bluetooth/Bluetooth Low-Energy (BLE).** Bluetooth is a low-power, short-range communications technology primarily designed for point-to-point communications between wireless devices in a hub-and-spoke topology. BLE (also known as Bluetooth Smart or Bluetooth 4.0+) devices consume significantly less power than Bluetooth devices and can access the internet directly through 6LoWPAN connectivity.

    - **Internet Protocol version 6 (IPv6) over Low-Power Wireless Personal Area Networks (6LoWPAN).** 6LoWPAN allows IPv6 (discussed in sections 2.1 and 2.1.1) traffic to be carried over low-power wireless mesh networks. 6LoWPAN is designed for nodes and applications requiring wireless internet connectivity at relatively low data rates in small form factors, such as smart light bulbs and smart meters.

    - **Wi-Fi/802.11.** The Institute of Electrical and Electronics Engineers (IEEE) defines the 802 LAN protocol standards. 802.11 is the set of standards used for Wi-Fi networks typically operating in the 2.4GHz and 5GHz frequency bands. Some of the most common implementations today include:

    - 802.11n (labeled Wi-Fi 4 by the Wi-Fi Alliance), which operates on both 2.4GHz and 5GHz bands at ranges from 54 megabits per second (Mbit/s) to 600Mbit/s

    - 802.11ac (Wi-Fi 5), which operates on the 5GHz band at ranges from 433Mbit/s to 3.46 gigabits per second (Gbit/s)

    - 802.11ax (Wi-Fi 6), which operates on the 2.4GHz and 5GHz bands (as well as all bands between 1 and 6GHz, when they become available for 802.11 use) at ranges up to 11Gbit/s

    - **Z-Wave.** Z-Wave is a low-energy wireless mesh network protocol primarily used for home automation applications such as smart appliances, lighting control, security systems, smart thermostats, windows and locks, and garage doors.

    - **ZigBee/802.14.** ZigBee is a low-cost, low-power wireless mesh network protocol based on the IEEE 802.15.4 standard. ZigBee is the dominant protocol in the low-

power networking market, with a large installed base in industrial environments and smart home products.

- **Low-power WAN (LP-WAN) and other wireless WAN (WWAN):**
    - **Narrowband IoT (NB-IoT).** NB-IoT provides low cost, long battery life, and high connection density for indoor applications. It uses a subset of the LTE standard in the 200 kilohertz (kHz) range.
    - **LoRa.** The LoRa Alliance is driving the Long-Range Wide-Area Network (LoRaWAN) protocol as the open global standard for secure, carrier-grade IoT low-power wide-area (LPWA) connectivity, primarily for large-scale public networks with a single operator.
    - **Sigfox.** Sigfox provides subscription-based global cellular LPWA connectivity for IoT devices. The Sigfox network relies on Ultra Narrowband (UNB) modulation and operates in unlicensed sub-GHz frequency bands.
    - **Worldwide Interoperability for Microwave Access (WiMAX).** WiMAX is a family of wireless broadband communications standards based on the IEEE 802.16 standards. WiMAX applications include portable mobile broadband connectivity, smart grids and metering, and internet failover for business continuity.

Identity of Things (IDoT) refers to identity and access management (IAM) solutions for the IoT. These solutions must be able to manage human-to-device, device-to-device, and/or device-to-service/system IAM by:

- Establishing a naming system for IoT devices.
- Determining an identity lifecycle for IoT devices, ensuring that it can be modified to meet the projected lifetime of IoT devices.
- Creating a well-defined process for registering IoT devices. The type of data that the device will be transmitting and receiving should shape the registration process.
- Defining security safeguards for data streams from IoT devices.
- Outlining well-defined authentication and authorization processes for admin local access to connected devices.
- Creating safeguards for protecting different types of data and making sure to create privacy safeguards for personally identifiable information (PII).

Though the IoT opens the door for innovative new approaches and services in all industries, it also presents new cybersecurity risks. According to research conducted by the Palo Alto Networks Unit 42 threat intelligence team, the general security posture of IoT devices is declining, leaving organizations vulnerable to new IoT-targeted malware as well as older attack techniques that IT teams have long forgotten. Key findings include:

- **IoT devices are unencrypted and unsecured.** Ninety-eight percent of all IoT device traffic is unencrypted, exposing personal and confidential data on the network. Attackers who have successfully bypassed the first line of defense (most frequently via phishing attacks) and established C2 are able to listen to unencrypted network traffic, collect personal or confidential information, and then exploit that data for profit on the dark web.

  Fifty-seven percent of IoT devices are vulnerable to medium- or high-severity attacks, making IoT low-hanging fruit for attackers. Because of the generally low patch level of IoT assets, the most frequent attacks are exploits via long-known vulnerabilities and password attacks using default device passwords.

- **Internet of Medical Things (IoMT) devices are running outdated software.** Eighty-three percent of medical imaging devices run on unsupported operating systems, which is a 56 percent jump from 2018. The increase is a result of the Windows 7 operating system reaching its end of life. This general decline in security posture opens the door for new attacks, such as cryptojacking (which increased from 0 percent in 2017 to 5 percent in 2019) and brings back long-forgotten attacks such as Conficker, which IT environments had previously been immune to for a long time.

  The IoMT devices with the most security issues are imaging systems, which represent a critical part of the clinical workflow. For healthcare organizations, 51 percent of threats involve imaging devices, which disrupts the quality of care and allows attackers to exfiltrate patient data stored on these imaging devices.

- **Healthcare organizations are displaying poor network security hygiene.** Seventy-two percent of healthcare VLANs mix IoT and IT assets, allowing malware to spread from users' computers to vulnerable IoT devices on the same network. There is a 41 percent rate of attacks exploiting device vulnerabilities as IT-borne attacks scan through network-connected devices in an attempt to exploit known weaknesses. There is an ongoing shift from IoT botnets conducting denial-of-service attacks to more sophisticated attacks targeting patient identities, corporate data, and monetary profit via ransomware.

- **IoT-focused cyberattacks are targeting legacy protocols.** There is an evolution of threats targeting IoT devices using new techniques, such as peer-to-peer C2 communications and wormlike features for self-propagation. Attackers recognize the vulnerability of decades-old legacy operational technology (OT) protocols, such as Digital Imaging and Communications in Medicine (DICOM), and can disrupt critical business functions in the organization.

2.0 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Fill in the Blank.** A _____ sends data packets to destination networks along a network path using logical addresses.
2. **Multiple Choice.** Which option is an example of a static routing protocol? (Choose one.)
    a) Open Shortest Path First (OSPF)
    b) Border Gateway Protocol (BGP)
    c) Routing Information Protocol (RIP)
    d) split horizon
3. **Multiple Choice.** Which three options are dynamic routing protocols? (Choose three.)
    a) distance-vector
    b) path-vector
    c) link-state
    d) point-to-point
4. **True or False.** The internet is an example of a wide-area network (WAN).
5. **Fill in the Blank.** The _____ is a distributed, hierarchical internet database that maps FQDNs to IP addresses.

2.1 Physical, Logical, and Virtual Addressing

Physical, logical, and virtual addressing in computer networks requires a basic understanding of decimal (base10), binary (base2), and hexadecimal (base16) numbering (see Table 2-1).

The decimal (base10) numbering system is, of course, what we all are taught in school. It comprises the numerals 0 through 9. After the number 9, we add a digit ("1") in the "tens" position and begin again at zero in the "ones" position, thereby creating the number 10. Humans use the decimal numbering system because we have ten fingers, so a base10 numbering system is easiest for humans to understand.

Table 2-1

Decimal, Hexadecimal, and Binary Notation

Decimal	Hexadecimal	Binary
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

A binary (base2) numbering system comprises only two digits: 1 ("on") and 0 ("off"). Binary numbering is used in computers and networking because they use electrical transistors (rather than fingers) to count. The basic function of a transistor is a gate: When electrical current is

present, the gate is closed (“1” or “on”). When no electrical current is present, the gate is open (“0” or “off”). With only two digits, a binary numbering system increments to the next position more frequently than a decimal numbering system. For example, the decimal number one is represented in binary as “1,” number two is represented as “10,” number three is represented as “11,” and number four is represented as “100.”

A hexadecimal (base16) numbering system comprises 16 digits (0 through 9 and A through F). Hexadecimal numbering is used because it is more convenient to represent a byte (which consists of 8 bits) of data as two digits in hexadecimal, rather than eight digits in binary. The decimal numbers 0 through 9 are represented as in hexadecimal “0” through “9,” respectively. However, the decimal number 10 is represented in hexadecimal as “A,” the number 11 is represented as “B,” the number 12 is represented as “C,” the number 13 is represented as “D,” the number 14 is represented as “E,” and the number 15 is represented as “F.” The number 16 then increments to the next numeric position, represented as “10.”

The physical address of a network device, known as a *media access control* (MAC) address (also referred to as a burned-in address [BIA] or hardware address), is used to forward traffic on a local network segment. The MAC address is a unique 48-bit identifier assigned to the network adapter of a device. If a device has multiple NICs, each NIC must have a unique MAC address. The MAC address is usually assigned by the device manufacturer and is stored in the device read-only memory (ROM) or firmware. MAC addresses are typically expressed in hexadecimal format with a colon or hyphen separating each 8-bit section. An example of a 48-bit MAC address is:

00:40:96:9d:68:16

The logical address of a network device, such as an IP address, is used to route traffic from one network to another. An IP address is a unique 32-bit or 128-bit (IPv4 and IPv6, respectively) address assigned to the NIC of a device. If a device has multiple NICs, each NIC may be assigned a unique IP address, or multiple NICs may be assigned a virtual IP address to enable bandwidth aggregation or failover capabilities. IP addresses are statically or dynamically (most commonly using *Dynamic Host Configuration Protocol*, or DHCP) assigned, typically by a network administrator or network service provider (NSP). IPv4 addresses are usually expressed in dotted decimal notation with a dot separating each decimal section (known as an *octet*). An example of an IPv4 address is:

192.168.0.1

IPv6 addresses are typically expressed in hexadecimal format (32 hexadecimal numbers grouped into eight blocks) with a colon separating each block of four hexadecimal digits (known as a *hextet*). An example of an IPv6 address is:

2001:0db8:0000:0000:0008:0800:200c:417a

IPv4 and IPv6 addressing is explained further in Section 2.1.1.

Address Resolution Protocol (ARP) translates a logical address, such as an IP address, to a physical MAC address. *Reverse Address Resolution Protocol* (RARP) translates a physical MAC address to a logical address.

Key Terms

A *media access control* (MAC) address is a unique 48-bit or 64-bit identifier assigned to a network interface card (NIC) for communications at the Data Link layer of the OSI model (discussed in Section 2.2.1).

Dynamic Host Configuration Protocol (DHCP) is a network management protocol that dynamically assigns (leases) IP addresses and other network configuration parameters (such as *default gateway* and DNS information) to devices on a network.

A *default gateway* is a network device, such as a router or switch, to which an endpoint sends network traffic when a specific destination IP address is not specified by an application or service, or when the endpoint does not know how to reach a specified destination.

An *octet* is a group of 8 bits in a 32-bit IPv4 address.

A *hextet* is a group of four 4-bit hexadecimal digits in a 128-bit IPv6 address.

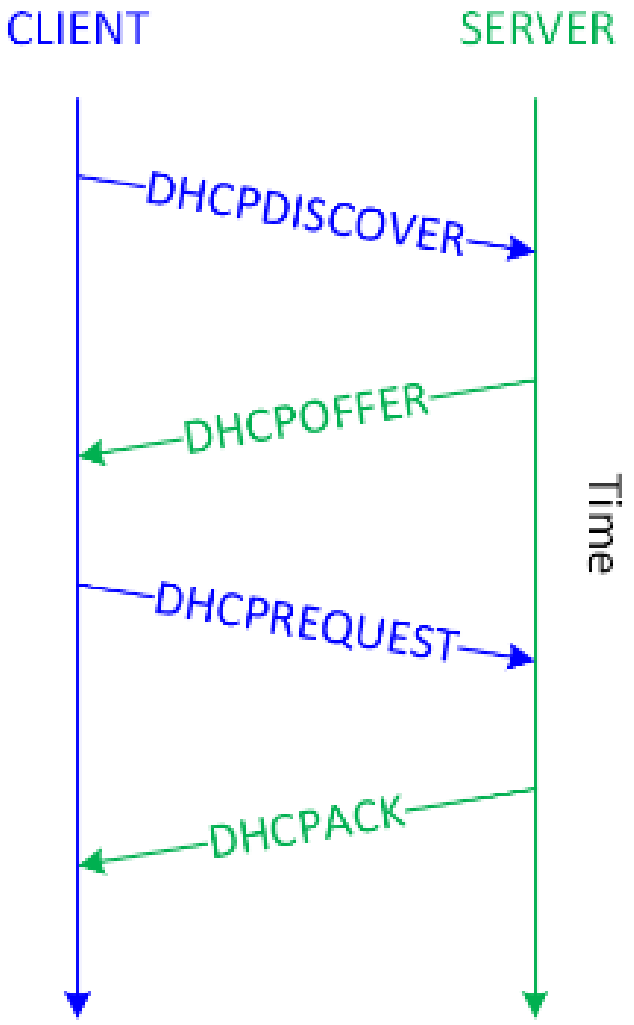
Address Resolution Protocol (ARP) translates a logical address, such as an IP address, to a physical MAC address. *Reverse Address Resolution Protocol* (RARP) translates a physical MAC address to a logical address.

DHCP is a network management protocol used to dynamically assign IP addresses to devices that do not have a statically assigned (manually configured) IP address on a TCP/IP network. Bootstrap Protocol (BOOTP) is a similar network management protocol that is commonly used on Unix and Linux TCP/IP networks. When a network-connected device that does not have a statically assigned IP address is powered on, the DHCP client software on the device broadcasts a DHCPDISCOVER message on UDP port 67. When a DHCP server on the same subnet (or a different subnet if a DHCP Helper or DHCP Relay Agent is configured) as the client receives the DHCPDISCOVER message, it reserves an IP address for the client and sends a DHCPOFFER message to the client on UDP port 68. The DHCPOFFER message contains the MAC address of the client, the IP address that is being offered, the subnet mask, the lease duration, and the IP address of the DHCP server that made the offer. When the client receives the DHCPOFFER, it

broadcasts a DHCPREQUEST message on UDP port 67, requesting the IP address that was offered. A client may receive DHCPOFFER messages from multiple DHCP servers on a subnet but can accept only one offer. When the DHCPREQUEST message is broadcast, the other DHCP servers that sent an offer that was not requested (in effect, accepted) in the DHCPREQUEST message will withdraw their offers. Finally, when the correct DHCP server receives the DHCPREQUEST message, it sends a DHCPACK (acknowledgment) message on UDP port 68, and the IP configuration process is completed (see Figure 2-1).

Figure 2-1

DHCP operation



Network address translation (NAT) virtualizes IP addresses by mapping private, non-routable IP addresses (discussed in Section 2.1.1) that are assigned to internal network devices to public IP addresses when communication across the internet is required. NAT is commonly implemented on firewalls and routers to conserve public IP addresses.

Key Terms

Network address translation (NAT) virtualizes IP addresses by mapping private, non-routable IP addresses assigned to internal network devices to public IP addresses.

2.1.1 IP addressing basics

Data packets are routed over a Transmission Control Protocol/Internet Protocol (TCP/IP) network using IP addressing information. IPv4, which is the most widely deployed version of IP, consists of a 32-bit logical IP address. The first four bits in an octet are known as the *high-order* bits; the first bit in the octet is referred to as the *most significant* bit. The last four bits in an octet are known as the *low-order* bits; the last bit in the octet is referred to as the *least significant* bit.

As shown in Table 2-2, each bit position represents its value if the bit is "on" (1); otherwise, the bit's value is zero ("off" or 0).

Key Terms

The first four bits in a 32-bit IPv4 address octet are referred to as the *high-order* bits.

The last four bits in a 32-bit IPv4 address octet are referred to as the *low-order* bits.

The first bit in a 32-bit IPv4 address octet is referred to as the *most significant* bit.

The last bit in a 32-bit IPv4 address octet is referred to as the *least significant* bit.

Table 2-2

Bit Position Values in an IPv4 Address

High-order bits				Low-order bits			
128	64	32	16	8	4	2	1

Each octet contains an 8-bit number with a value of 0 to 255. Table 2-3 shows a partial list of octet values in binary notation.

Table 2-3

Binary Notation of Octet Values

Decimal	Binary	Decimal	Binary	Decimal	Binary
255	1111 1111	172	1010 1100	64	0100 0000
254	1111 1110	170	1010 1010	32	0010 0000
253	1111 1101	160	1010 0000	16	0001 0000
252	1111 1100	150	1001 0110	8	0000 1000
251	1111 1011	140	1000 1100	7	0000 0111
250	1111 1010	130	1000 0010	6	0000 0110
249	1111 1001	128	1000 0000	5	0000 0101
248	1111 1000	120	0111 1000	4	0000 0100
224	1110 0000	110	0110 1110	3	0000 0011
200	1100 1000	100	0110 0100	2	0000 0010
192	1100 0000	96	0110 0000	1	0000 0001
180	1011 0100	90	0101 1010	0	0000 0000

The five IPv4 address classes (indicated by the high-order bits) are shown in Table 2-4.

Table 2-4

IP Address Classes

Class	Purpose	High-Order Bits	Address Range	Max. # of Hosts
A	Large networks	0	1 to 126	16,777,214
B	Medium-size networks	10	128 to 191	65,534
C	Small networks	110	192 to 223	254
D	Multicast	1110	224 to 239	–
E	Experimental	1111	240 to 254	–

The address range 127.0.0.1 to 127.255.255.255 is a loopback network used for testing and troubleshooting. Packets sent to a loopback (or localhost) address – such as 127.0.0.1 – are immediately routed back to the source device.

A *subnet mask* is a number that hides the network portion of an IPv4 address, leaving only the host portion of the IP address. The network portion of a subnet mask is represented by

contiguous “on” (1) bits beginning with the most significant bit. For example, in the subnet mask 255.255.255.0, the first three octets represent the network portion and the last octet represents the host portion of an IP address. Recall that the decimal number 255 is represented in binary notation as 1111 1111 (refer to Table 2-2).

Key Terms

A *subnet mask* is a number that hides the network portion of an IPv4 address, leaving only the host portion of the IP address.

The default (or standard) subnet masks for Class A, B, and C networks are:

Class A: 255.0.0.0

Class B: 255.255.0.0

Class C: 255.255.255.0

Several IPv4 address ranges are reserved for use in private networks and are not routable on the internet, including:

10.0.0.0–10.255.255.255 (Class A)

172.16.0.0–172.31.255.255 (Class B)

192.168.0.0–192.168.255.255 (Class C)

The 32-bit address space of an IPv4 address limits the total number of unique public IP addresses to about 4.3 billion. The widespread use of NAT (discussed in Section 2.1) delayed the inevitable depletion of IPv4 addresses, but, as of 2018, the pool of available IPv4 addresses that can be assigned to organizations had officially been depleted. (A small pool of IPv4 addresses was reserved by each regional internet registry to facilitate the transition to IPv6.) IPv6 addresses, which use a 128-bit hexadecimal address space providing about 3.4×10^{38} (340 hundred undecillion) unique IP addresses, were created to replace IPv4 when the IPv4 address space was exhausted.

IPv6 addresses consist of 32 hexadecimal numbers grouped into eight hextets of four hexadecimal digits, separated by a colon. A hexadecimal digit is represented by 4 bits (refer to Table 2-1), so each hextet is 16 bits (four 4-bit hexadecimal digits), and eight 16-bit hextets equals 128 bits.

An IPv6 address is further divided into two 64-bit segments: the first (also referred to as the "top" or "upper") 64 bits represent the network part of the address, and the last (also referred to as the "bottom" or "lower") 64 bits represent the node or interface part of the address. The network part is further subdivided into a 48-bit global network address and a 16-bit subnet. The node or interface part of the address is based on the MAC address (discussed in Section 2.1) of the node or interface.

The basic format for an IPv6 address is:

xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx

where x represents a hexadecimal digit (0–f).

This is an example of an IPv6 address:

2001:0db8:0000:0000:0008:0800:200c:417a

The Internet Engineering Task Force (IETF) has defined several rules to simplify an IPv6 address:

- Leading zeros in an individual hextet may be omitted, but each hextet must have at least one hexadecimal digit, except as noted in the next rule. Applying this rule to the previous example yields this result: 2001:db8:0:0:8:800:200c:417a.
- Two colons (::) may be used to represent one or more groups of 16 bits of zeros, and leading or trailing zeroes in an address; however, the two colons (::) may appear only once in an IPv6 address. Applying this rule to the previous example yields this result: 2001:db8::8:800:200c:417a.
- In mixed IPv4 and IPv6 environments, the form x:x:x:x:x;x:d.d.d.d may be used, in which x represents the six high-order 16-bit hextets of the address and "d" represents the four low-order 8-bit octets (in standard IPv4 notation) of the address. For example, 0db8:0:0:0:0:FFFF:129.144.52.38 is a valid IPv6 address. Application of the previous two rules to this example yields this result: db8::ffff:129.144.52.38.

IPv6 security features are specified in Request for Comments (RFC) 7112 and include techniques to prevent fragmentation exploits in IPv6 headers and implementation of Internet Protocol Security (IPsec, discussed in Section 2.3.4.6) at the Network layer of the OSI model (discussed in Section 2.2.1).

2.1.2 Introduction to subnetting

Subnetting is a technique used to divide a large network into smaller, multiple subnetworks by segmenting an IP address into two parts: the network and the host. Subnetting can be used to

limit network traffic or limit the number of devices that are visible to, or can connect to, each other. Subnetting is also commonly used to logically organize a network; for example, by assigning different departments or geographic locations to different subnets. As shown in Table 2-5, an organization that has been assigned a Class B public IP address (172.168.0.0) could use this address space as a single network with up to 65,534 hosts on the network using the default subnet mask (255.255.0.0) for a Class B network. However, such a large flat network would be difficult to manage. Instead, the organization may choose to use a 24-bit subnet mask (255.255.255.0) to separate the 172.168.0.0 network into as many as 256 different networks, each with up to 254 hosts on the subnet, and assign each subnet to a different geographic location representing the organization's various offices. Routers examine IP addresses and

Key Terms

Subnetting is a technique used to divide a large network into smaller subnetworks.

subnet values (called masks) and determine whether to forward packets between networks. With IP addressing, the subnet mask is a required element.

Table 2-5

An example of organizing a network by geographic location using a 16-bit subnet mask.

Subnet	Location
172.168.1.0	Atlanta
172.168.2.0	Boston
172.168.3.0	Chicago
172.168.4.0	Dallas
172.168.5.0	New York
172.168.6.0	San Francisco
172.168.7.0	Seattle

For a Class C IPv4 address, there are 254 possible node (or host) addresses (2^8 or 256 potential addresses, but two addresses are lost for each network: one for the base network address and the other for the broadcast address). A typical Class C network uses a default 24-bit subnet mask (255.255.255.0). This subnet mask value identifies the network portion of an IPv4 address, with the first three octets being all ones (11111111 in binary notation, 255 in decimal notation). The mask displays the last octet as zero (00000000 in binary notation). For a Class C

IPv4 address with the default subnet mask, the last octet is where the node-specific values of the IPv4 address are assigned.

For example, in a network with an IPv4 address of 192.168.1.0 and a mask value of 255.255.255.0, the network portion of the address is 192.168.1, and there are 254 node addresses (192.168.1.1 through 192.168.1.254) available. Remember, the first address (192.168.1.0) is the base network, and the last address (192.168.1.255) is the broadcast address.

Class A and Class B IPv4 addresses use smaller mask values and support larger numbers of nodes than Class C IPv4 addresses for their default address assignments. Class A networks use a default 8-bit (255.0.0.0) subnet mask, which provides a total of more than 16 million (256 x 256 x 256) available IPv4 node addresses. Class B networks use a default 16-bit (255.255.0.0) subnet mask, which provides a total of 65,534 (256 x 256 minus the network address and the broadcast address) available IPv4 node addresses.

Unlike subnetting, which divides an IPv4 address along an arbitrary (default) classful 8-bit boundary (8 bits for a Class A network, 16 bits for a Class B network, 24 bits for a Class C network), *classless inter-domain routing* (CIDR) allocates address space on any address bit boundary (known as *variable-length subnet masking*, or VLSM). For example, using CIDR, a Class A network could be assigned a 24-bit mask (255.255.255.0, instead of the default 8-bit 255.0.0.0 mask) to limit the subnet to only 254 addresses, or a 23-bit mask (255.255.254.0) to limit the subnet to 512 addresses.

CIDR is used to reduce the size of routing tables on internet routers by aggregating multiple contiguous network prefixes (known as *supernetting*), and it also helps slow the depletion of public IPv4 addresses (discussed in Section 2.1.1).

Key Terms

Classless inter-domain routing (CIDR) is a method for allocating IP addresses and IP routing that replaces classful IP addressing (for example, Class A, B, and C networks) with classless IP addressing.

Variable-length subnet masking (VLSM) is a technique that enables IP address spaces to be divided into different sizes.

Supernetting aggregates multiple contiguous smaller networks into a larger network to enable more efficient internet routing.

An IP address can be represented with its subnet mask value, using "netbit" or CIDR notation. A netbit value represents the number of ones in the subnet mask and is displayed after an IP address, separated by a forward slash. For example, 192.168.1.0/24 represents a subnet mask consisting of 24 ones:

11111111.11111111.11111111.00000000 (in binary notation)

or

255.255.255.0 (in decimal notation)

2.1 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Multiple Choice.** Which option is an example of a logical address? (Choose one.)
    a) IP address
    b) hardware address
    c) MAC address
    d) burned-in address
2. **Fill in the Blank.** An IPv4 address consists of four __-bit octets.
3. **Fill in the Blank.** _____ is a technique used to divide a large network into smaller subnetworks by segmenting an IPv4 address into network and host portions.

2.2 Packet Encapsulation and Lifecycle

In a *circuit-switched* network, a dedicated physical circuit path is established, maintained, and terminated between the sender and receiver across a network for each communications session. Before the development of the internet, most communications networks, such as telephone company networks, were circuit-switched. As discussed in Section 2.0.1, the internet is a *packet-switched* network comprising hundreds of millions of routers and billions of servers and user endpoints. In a packet-switched network, devices share bandwidth on communications links to transport packets between a sender and a receiver across a network. This type of network is more resilient to error and congestion than circuit-switched networks.

An application that needs to send data across the network (for example, from a server to a client computer) first creates a block of data and sends it to the TCP stack on the server. The TCP stack places the block of data into an output buffer on the server and determines the maximum segment size (MSS) of individual TCP blocks (*segments*) permitted by the server operating system. The TCP stack then divides the data blocks into appropriately sized segments (for example, 1460 bytes), adds a TCP header, and sends the segment to the IP stack on the server. The IP stack adds source (sender) and destination (receiver) IP addresses to the TCP segment (which is now called an IP packet) and notifies the server operating system that it has an outgoing message that is ready to be sent across the network. When the server operating system is ready, the IP packet is sent to the network adapter, which converts the IP packet to bits and sends the message across the network.

On their way to the destination computer, the packets typically traverse several network and security devices (such as switches, routers, and firewalls) before reaching the destination computer, where the encapsulation process described above is reversed.

Key Terms

In a *circuit-switched network*, a dedicated physical circuit path is established, maintained, and terminated between the sender and the receiver across a network for each communications session.

In a *packet-switched network*, devices share bandwidth on communications links to transport packets between the sender and the receiver across a network.

A TCP *segment* is a *protocol data unit* (PDU) defined at the Transport layer of the OSI model.

A *protocol data unit* (PDU) is a self-contained unit of data (consisting of user data or control information and network addressing).

2.2.1 The OSI and TCP/IP models

The *Open Systems Interconnection* (OSI) and *Transmission Control Protocol/Internet Protocol* (TCP/IP) models define standard protocols for network communication and interoperability. Using a layered approach, the OSI and TCP/IP models:

- Clarify the general functions of communications processes
- Reduce complex networking processes to simpler sublayers and components
- Promote interoperability through standard interfaces

- Enable vendors to change individual features at a single layer rather than rebuild the entire protocol stack
- Facilitate logical troubleshooting

Defined by the International Organization for Standardization (ISO – not an acronym but the adopted organizational name from the Greek *isos*, meaning "equal"), the OSI model consists of seven layers:

- **Application (Layer 7 or L7).** This layer identifies and establishes availability of communication partners, determines resource availability, and synchronizes communication. End-user software, such as web browsers and email clients, operate at the Application layer. Protocols that function at the Application layer include:
    - **File Transfer Protocol (FTP).** Used to copy files from one system to another on TCP ports 20 (the data port) and 21 (the control port)
    - **Hypertext Transfer Protocol (HTTP).** Used for communication between web servers and web browsers on TCP port 80
    - **Hypertext Transfer Protocol Secure (HTTPS).** Used for Secure Sockets Layer/Transport Layer Security (SSL/TLS) encrypted communications between web servers and web browsers on TCP port 443 (and other ports, such as 8443)
    - **Internet Message Access Protocol (IMAP).** A store-and-forward electronic mail protocol that allows an email client to access, manage, and synchronize email on a remote mail server on TCP and UDP port 143
    - **Post Office Protocol Version 3 (POP3).** An email retrieval protocol that allows an email client to access email on a remote mail server on TCP port 110
    - **Simple Mail Transfer Protocol (SMTP).** Used to send and receive email across the internet on TCP/UDP port 25
    - **Simple Network Management Protocol (SNMP).** Used to collect network information by polling stations and sending traps (or alerts) to a management station on TCP/UDP ports 161 (agent) and 162 (manager)
    - **Telnet.** Provides terminal emulation for remote access to system resources on TCP/UDP port 23
- **Presentation (Layer 6 or L6).** This layer provides coding and conversion functions (such as data representation, character conversion, data compression, and data encryption) to

ensure that data sent from the Application layer of one system is compatible with the Application layer of the receiving system. Protocols that function at the Presentation layer include:

- **American Standard Code for Information Interchange (ASCII).** A character-encoding scheme based on the English alphabet, consisting of 128 characters
- **Extended Binary-Coded Decimal Interchange Code (EBCDIC).** An 8-bit character-encoding scheme largely used on mainframe and midrange computers
- **Graphics Interchange Format (GIF).** A bitmap image format that allows up to 256 colors and is suitable for images or logos (but not photographs)
- **Joint Photographic Experts Group (JPEG).** A photographic compression method used to store and transmit photographs
- **Motion Picture Experts Group (MPEG).** An audio and video compression method used to store and transmit audio and video files

- **Session (Layer 5 or L5).** This layer manages communication sessions (service requests and service responses) between networked systems, including connection establishment, data transfer, and connection release. Protocols that function at the Session layer include:
  - **Network File System (NFS).** Facilitates transparent user access to remote resources on a Unix-based TCP/IP network
  - **Remote Procedure Call (RPC).** A client-server network redirection protocol
  - **Secure Shell (SSH).** Establishes an encrypted tunnel between a client and a server
  - **Session Initiation Protocol (SIP).** An open signaling protocol standard for establishing, managing, and terminating real-time communications (such as voice, video, and text) over large IP-based networks
- **Transport (Layer 4 or L4).** This layer provides transparent, reliable data transport and end-to-end transmission control. The Transport layer receives data from the Session layer and breaks it into segments that can then be sent to the Network layer, then reassembles segments received from the Network layer and sends them to the Session layer. Specific Transport layer functions include:

- **Flow control.** Manages data transmission between devices by ensuring that the transmitting device doesn't send more data than the receiving device can process

- **Multiplexing.** Enables data from multiple applications to be simultaneously transmitted over a single physical link

- **Virtual circuit management.** Establishes, maintains, and terminates virtual circuits

- **Error checking and recovery.** Detects transmission errors and takes action to resolve any errors that occur; for example, requesting that data be retransmitted

TCP and UDP port numbers assigned to applications and services are defined at the Transport layer. Protocols that function at the Transport layer include:

- **Transmission Control Protocol (TCP).** A connection-oriented (a direct connection between network devices is established before data segments are transferred) protocol that provides reliable delivery (received segments are acknowledged and retransmission of missing or corrupted segments is requested) of data. TCP connections are established via a *three-way handshake*. The additional overhead associated with connection establishment, acknowledgment, and error correction means that TCP is generally slower than connectionless protocols such as User Datagram Protocol (UDP).

- **User Datagram Protocol (UDP).** A connectionless (a direct connection between network devices is not established before *datagrams* are transferred) protocol that provides best-effort delivery (received datagrams are not acknowledged and missing or corrupted datagrams are not requested) of data. UDP has no overhead associated with connection establishment, acknowledgment, sequencing, or error-checking and recovery. UDP is ideal for data that requires fast delivery provided that data is not sensitive to packet loss and does not require fragmentation. Applications that use UDP include Domain Name System (DNS), Simple Network Management Protocol (SNMP), and streaming audio or video.

- **Stream Control Transmission Protocol (SCTP).** A message-oriented protocol (similar to UDP) that ensures reliable, in-sequence transport with congestion control (similar to TCP).

- **Network (Layer 3 or L3).** This layer provides routing and related functions that enable data to be transported between systems on the same network or on interconnected networks. Routing protocols (discussed in Section 2.2.3) are defined at this layer. Logical addressing of devices on the network is accomplished at this layer using routed protocols such as Internet Protocol (IP). Routers operate at the Network layer of the OSI model.

- **Data Link (Layer 2).** This layer ensures that messages are delivered to the proper device across a physical network link. This layer also defines the networking protocol (for example, Ethernet) used to send and receive data between individual devices and formats messages from the layers listed above into frames for transmission, handles point-to-point synchronization and error control, and can perform link encryption. Switches typically operate at Layer 2 of the OSI model (although multilayer switches that operate at different layers also exist). The Data Link layer is further divided into two sublayers:

    - **Logical Link Control (LLC).** The LLC sublayer provides an interface for the MAC sublayer; manages the control, sequencing, and acknowledgment of frames being passed up to the Network layer or down to the Physical layer; and manages timing and *flow control*.

    - **Media access control (MAC).** The MAC sublayer is responsible for framing and performs error control using a *cyclic redundancy check* (CRC), identifies MAC addresses (discussed in Section 2.1), and controls media access.

- **Physical (Layer 1 or L1).** This layer sends and receives bits across the network medium (cabling or wireless links) from one device to another. It specifies the electrical, mechanical, and functional requirements of the network, including network topology, cabling and connectors, and interface types, as well as the process for converting bits to electrical (or light) signals that can be transmitted across the physical medium.

The TCP/IP model was originally developed by the U.S. Department of Defense (DoD) and actually preceded the OSI model. Whereas the OSI model is a theoretical model used to logically describe networking processes, the TCP/IP model defines actual networking requirements; for example, for frame construction. The TCP/IP model consists of four layers (see Figure 2-2):

- **Application (Layer 4 or L4).** This layer consists of network applications and processes, and it loosely corresponds to Layers 5 through 7 of the OSI model.

- **Transport (Layer 3 or L3).** This layer provides end-to-end delivery, and it corresponds to Layer 4 of the OSI model.
- **Internet (Layer 2 or L2).** This layer defines the IP datagram and routing, and it corresponds to Layer 3 of the OSI model.
- **Network Access (Layer 1 or L1).** Also referred to as the Link layer, this layer contains routines for accessing physical networks, and it corresponds to Layers 1 and 2 of the OSI model.

Key Terms

In TCP, a *three-way handshake* is used to establish a connection. For example, a PC initiates a connection with a server by sending a TCP SYN (Synchronize) packet. The server replies with a SYN ACK packet (Synchronize Acknowledgment). Finally, the PC sends an ACK or SYN-ACK-ACK packet acknowledging the server's acknowledgment, and data communication begins.

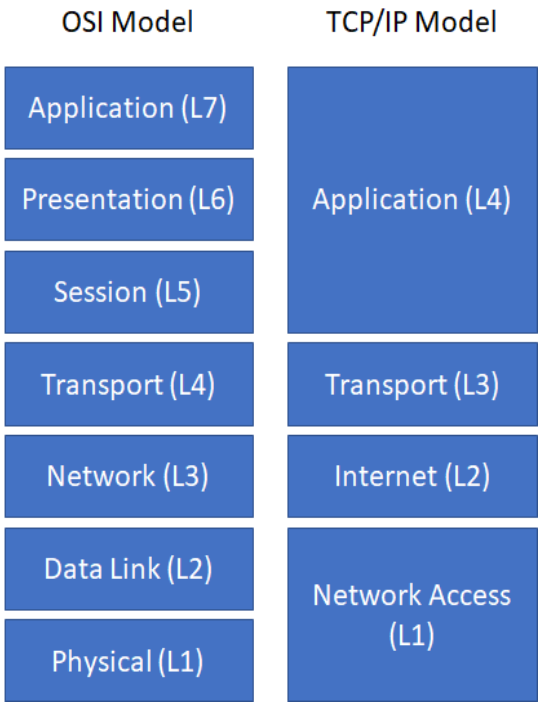
A UDP *datagram* is a PDU defined at the Transport layer of the OSI model.

Flow control monitors the flow of data between devices to ensure that a receiving device, which may not necessarily be operating at the same speed as the transmitting device, does not drop packets.

A *cyclic redundancy check* (CRC) is a checksum used to create a message profile. The CRC is recalculated by the receiving device. If the recalculated CRC doesn't match the received CRC, the packet is dropped, and a request to resend the packet is transmitted back to the device that sent the packet.

Figure 2-2

The OSI model and the TCP/IP model



2.2.2 Data encapsulation

In the OSI and TCP/IP models, data is passed from the highest layer (L7 in the OSI model, L4 in the TCP/IP model) downward through each layer to the lowest layer (L1 in the OSI model and the TCP/IP model). It is then transmitted across the network medium to the destination node, where it is passed upward from the lowest layer to the highest layer. Each layer communicates only with the adjacent layer immediately above and below it. This communication is achieved through a process known as *data encapsulation* (or *data hiding*), which wraps protocol information from the layer immediately above in the data section of the layer immediately below.

Key Terms

Data encapsulation (or *data hiding*) wraps protocol information from the (OSI or TCP/IP) layer immediately above in the data section of the layer below.

A protocol data unit (PDU) describes a unit of data at a particular layer of a protocol. For example, in the OSI model, a Layer 1 PDU is known as a bit, a Layer 2 PDU is known as a frame, a Layer 3 PDU is known as a packet, and a Layer 4 PDU is known as a segment or datagram. When a client or server application sends data across a network, a header (and trailer in the

case of Layer 2 frames) is added to each data packet from the adjacent layer below it as the data passes through the protocol stack. On the receiving end, the headers (and trailers) are removed from each data packet as it passes through the protocol stack to the receiving application.

2.2.3 Routed and routing protocols

Routed protocols, such as *Internet Protocol* (IP), address packets with routing information that enables those packets to be transported across networks using routing protocols. Internet Protocol (IP) is discussed further in Sections 2.1 and 2.1.1.

Key Terms

An *Internet Protocol (IP) address* is a 32-bit or 128-bit identifier assigned to a networked device for communications at the Network layer of the OSI model or the Internet layer of the TCP/IP model.

Routing protocols are defined at the Network layer of the OSI model (discussed in Section 2.2.1) and specify how routers communicate with one another on a network. Routing protocols can either be static or dynamic.

A static routing protocol requires that routes be created and updated manually on a router or other network device. If a static route is down, traffic can't be automatically rerouted unless an alternate route has been configured. Also, if the route is congested, traffic can't be automatically rerouted over the less congested alternate route. Static routing is practical only in very small networks or for very limited, special-case routing scenarios (for example, a destination that's used as a backup route or is reachable only via a single router). However, static routing has low bandwidth requirements (routing information isn't broadcast across the network) and some built-in security (users can route only to destinations that are specified in statically defined routes).

A dynamic routing protocol can automatically learn new (or alternate) routes and determine the best route to a destination. The routing table is updated periodically with current routing information. Dynamic routing protocols are further classified as:

- **Distance-vector.** A distance-vector protocol makes routing decisions based on two factors: the distance (hop count or other metric) and vector (the egress router interface). It periodically informs its peers and/or neighbors of topology changes. *Convergence*, the time required for all routers in a network to update their routing tables with the most current information (such as link status changes), can be a

significant problem for distance-vector protocols. Without convergence, some routers in a network may be unaware of topology changes, which causes the router to send traffic to an invalid destination. During convergence, routing information is exchanged between routers, and the network slows down considerably. Convergence can take several minutes in networks that use distance-vector protocols.

Routing Information Protocol (RIP) is an example of a distance-vector routing protocol that uses *hop count* as its routing metric. To prevent routing loops, in which packets effectively get stuck bouncing between various router nodes, RIP implements a hop limit of 15, which limits the size of networks that RIP can support. After a data packet crosses 15 router nodes (hops) between a source and a destination, the destination is considered unreachable. In addition to hop limits, RIP employs four other mechanisms to prevent routing loops:

- **Split horizon.** Prevents a router from advertising a route back out through the same interface from which the route was learned.
- **Triggered updates.** When a change is detected, the update gets sent immediately instead of waiting 30 seconds to send a RIP update.
- **Route poisoning.** Sets the hop count on a bad route to 16, which effectively advertises the route as unreachable.
- **Holddown timers.** Causes a router to start a timer when the router first receives information that a destination is unreachable. Subsequent updates about that destination will not be accepted until the timer expires. This timer also helps avoid problems associated with flapping. Flapping occurs when a route (or interface) repeatedly changes state (up, down, up, down) over a short period of time.

- **Link state.** A link-state protocol requires every router to calculate and maintain a complete map, or routing table, of the entire network. Routers that use a link-state protocol periodically transmit updates that contain information about adjacent connections, or link states, to all other routers in the network. Link-state protocols are compute-intensive, but they can calculate the most efficient route to a destination. They consider numerous factors, such as link speed, delay, load, reliability, and *cost* (an arbitrarily assigned weight or metric). Convergence occurs very rapidly (within seconds) with link-state protocols.

  Open Shortest Path First (OSPF) is an example of a link-state routing protocol that is often used in large enterprise networks. OSPF routes network traffic within a single

autonomous system (AS). OSPF networks are divided into areas identified by 32-bit area identifiers. Area identifiers can (but don't have to) correspond to network IP addresses and can duplicate IP addresses without conflicts.

- **Path vector.** A path-vector protocol is similar to a distance-vector protocol but without the scalability issues associated with limited hop counts in distance-vector protocols. Each routing table entry in a path-vector protocol contains path information that gets dynamically updated.

Key Terms

Convergence is the time required for all routers in a network to update their routing tables with the most current routing information about the network.

Hop count generally refers to the number of router nodes that a packet must pass through to reach its destination.

An *autonomous system* (AS) is a group of contiguous IP address ranges under the control of a single internet entity. Individual autonomous systems are assigned a 16-bit or 32-bit AS number (ASN) that uniquely identifies the network on the internet. ASNs are assigned by the Internet Assigned Numbers Authority (IANA).

Border Gateway Protocol (BGP) is an example of a path-vector protocol used between separate autonomous systems. BGP is the core protocol used by internet service providers (ISPs) and network service providers (NSPs), as well as on very large private IP networks.

2.2 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Multiple Choice.** The OSI model consists of how many layers? (Choose one.)
    a) four
    b) six
    c) seven
    d) nine
2. **Multiple Choice.** Which two protocols function at the Transport layer of the OSI model? (Choose two).
    a) Transmission Control Protocol (TCP)
    b) Internet Protocol (IP)
    c) User Datagram Protocol (UDP)
    d) Hypertext Transfer Protocol (HTTP)
3. **Fill in the Blank.** The Data Link layer of the OSI model is further divided into these two sublayers: _____ and_____.
4. **Multiple Choice.** Which four layers comprise the TCP/IP model? (Choose four.)
    a) Application
    b) Transport
    c) Physical
    d) Internet
    e) Network Access
5. **Fill in the Blank.** The process that wraps protocol information from the (OSI or TCP/IP) layer immediately above in the data section of the layer immediately below is known as _____.

2.3 Network Security Technologies

This section describes traditional network security technologies, including firewalls, intrusion detection systems (IDSs) and intrusion prevention systems (IPSs), web content filters, virtual private networks (VPNs), data loss prevention (DLP), unified threat management (UTM), and security information and event management (SIEM).

2.3.1 Firewalls

Firewalls have been a cornerstone of network security since the early days of the internet. A firewall is a hardware and/or software platform that controls the flow of traffic between a trusted network (such as a corporate LAN) and an untrusted network (such as the internet).

2.3.1.1 Packet filtering firewalls

First-generation *packet filtering* (also known as *port-based*) firewalls have the following characteristics:

- They operate up to Layer 4 (Transport layer) of the OSI model (discussed in Section 2.2.1) and inspect individual packet headers to determine source and destination IP address, protocol (TCP, UDP, ICMP), and port number.
- They match source and destination IP address, protocol, and port number information contained within each packet header to a corresponding rule on the firewall that designates whether the packet should be allowed, blocked, or dropped.
- They inspect and handle each packet individually, with no information about context or session.

2.3.1.2 Stateful packet inspection firewalls

Second-generation *stateful packet inspection* (also known as *dynamic packet filtering*) firewalls have the following characteristics:

- They operate up to Layer 4 (Transport layer) of the OSI model and maintain state information about the communication sessions that have been established between hosts on the trusted and untrusted networks.
- They inspect individual packet headers to determine source and destination IP address, protocol (TCP, UDP, and ICMP), and port number (during session establishment only) to determine if the session should be allowed, blocked, or dropped based on configured firewall rules.

- After a permitted connection is established between two hosts, the firewall creates and deletes firewall rules for individual connections on an as needed basis, thus effectively creating a tunnel that allows traffic to flow between the two hosts without further inspection of individual packets during the session.
- This type of firewall is very fast, but it is port-based and highly dependent on the trustworthiness of the two hosts because individual packets are not inspected after the connection is established.

2.3.1.3 Application firewalls

Third-generation *application* (also known as *Application layer gateways*, *proxy-based*, and *reverse-proxy*) firewalls have the following characteristics:

- They operate up to Layer 7 (Application layer) of the OSI model and control access to specific applications and services on the network.
- They proxy network traffic rather than permit direct communication between hosts. Requests are sent from the originating host to a proxy server, which analyzes the contents of the data packets and, if permitted, sends a copy of the original data packets to the destination host.
- They inspect Application layer traffic and thus can identify and block specified content, malware, exploits, websites, and applications or services using hiding techniques such as encryption and non-standard ports. Proxy servers can also be used to implement strong user authentication and web application filtering and to mask the internal network from untrusted networks. However, proxy servers have a significant negative impact on the overall performance of the network.

2.3.2 Intrusion detection and intrusion prevention systems

Intrusion detection systems (IDSs) and intrusion prevention systems (IPSs) provide real-time monitoring of network traffic and perform deep-packet inspection and analysis of network activity and data. Unlike traditional packet filtering and stateful packet inspection firewalls that examine only packet header information, an IDS/IPS examines both the packet header and the payload of network traffic. The IDS/IPS attempts to match known-bad, or malicious, patterns (or signatures) found within inspected packets. An IDS/IPS is typically deployed to detect and block exploits of software vulnerabilities on target networks.

The primary difference between an IDS and an IPS is that an IDS is considered to be a passive system, whereas an IPS is an active system. An IDS monitors and analyzes network activity and

provides alerts to potential attacks and vulnerabilities on the network, but it doesn't perform any preventive action to stop an attack. An IPS, however, performs all of the same functions as an IDS but also automatically blocks or drops suspicious, pattern-matching activity on the network in real time. However, an IPS has some disadvantages, including:

- It must be placed inline along a network boundary and is thus directly susceptible to attack itself.
- False alarms must be properly identified and filtered to avoid inadvertently blocking authorized users and applications. A false positive occurs when legitimate traffic is improperly identified as malicious traffic. A false negative occurs when malicious traffic is improperly identified as legitimate traffic.
- It may be used to deploy a denial-of-service (DoS) attack by flooding the IPS, thus causing it to block connections until no connection or bandwidth is available.

IDSs and IPSs can also be classified as knowledge-based (or signature-based) or behavior-based (or statistical anomaly-based) systems:

- A knowledge-based system uses a database of known vulnerabilities and attack profiles to identify intrusion attempts. These types of systems have lower false-alarm rates than behavior-based systems but must be continually updated with new attack signatures to be effective.
- A behavior-based system uses a baseline of normal network activity to identify unusual patterns or levels of network activity that may be indicative of an intrusion attempt. These types of systems are more adaptive than knowledge-based systems and may therefore be more effective in detecting previously unknown vulnerabilities and attacks, but they have a much higher false-positive rate than knowledge-based systems.

2.3.3 Web content filters

Web content filters are used to restrict the internet activity of users on a network. Web content filters match a web address (*uniform resource locator*, or URL) against a database of websites, which is typically maintained by the individual security vendors that sell the web content filters and is provided as a subscription-based service. Web content filters attempt to classify websites based on broad categories that are either allowed or blocked for various groups of users on the network. For example, the marketing and human resources departments may have access to social media sites such as Facebook and LinkedIn for legitimate online marketing and recruiting activities, while other users are blocked. Typical website categories include:

- Gambling and online gaming
- Hacking
- Hate crimes and violence
- Pornography
- Social media
- Web-based email

Key Terms

A *uniform resource locator* (URL) is a unique reference (or address) to an internet resource, such as a webpage.

These sites lower individual productivity but also may be prime targets for malware that users may unwittingly become victims of via drive-by downloads. Certain sites may also create liabilities in the form of sexual harassment or racial discrimination suits for organizations that fail to protect other employees from being exposed to pornographic or hate-based websites.

Organizations may elect to implement these solutions in a variety of modes to either block content, warn users before they access restricted sites, or log all activity. The disadvantage of blocking content is that false positives require the user to contact a security administrator to allow access to websites that have been improperly classified and blocked or need to be accessed for a legitimate business purpose.

2.3.4 Virtual private networks

A virtual private network (VPN) creates a secure, encrypted connection (or tunnel) across the internet back to an organization's network. VPN client software is typically installed on mobile endpoints, such as laptop computers and smartphones, to extend a network beyond the physical boundaries of the organization. The VPN client connects to a VPN server, such as a firewall, router, or VPN appliance (or concentrator). After a VPN tunnel is established, a remote user can access network resources – such as file servers, printers, and Voice over IP (VoIP) phones – in the same way as if they were physically located in the office.

2.3.4.1 Point-to-Point Tunneling Protocol

Point-to-Point Tunneling Protocol (PPTP) is a basic VPN protocol that uses Transmission Control Protocol (TCP) port 1723 to establish communication with the VPN peer and then creates a

Generic Routing Encapsulation (GRE) tunnel that transports encapsulated *Point-to-Point Protocol* (PPP) packets between the VPN peers. Although PPTP is easy to set up and is considered to be very fast, it is perhaps the least secure of the various VPN protocols. It is commonly used with either the *Password Authentication Protocol* (PAP), *Challenge-Handshake Authentication Protocol* (CHAP), or *Microsoft Challenge-Handshake Authentication Protocol versions 1* and *2* (MS-CHAP v1/v2), all of which have well-known security vulnerabilities, to authenticate tunneled PPP traffic. The *Extensible Authentication Protocol Transport Layer Security* (EAP-TLS) provides a more secure authentication protocol for PPTP but requires a *public key infrastructure* (PKI) and is therefore more difficult to set up.

Key Terms

Generic Routing Encapsulation (GRE) is a tunneling protocol developed by Cisco Systems that can encapsulate various Network layer protocols inside virtual point-to-point links.

Point-to-Point Protocol (PPP) is a Layer 2 (Data Link) protocol used to establish a direct connection between two nodes.

Password Authentication Protocol (PAP) is an authentication protocol used by PPP to validate users with an unencrypted password.

Microsoft Challenge-Handshake Authentication Protocol (MS-CHAP) is used to authenticate Microsoft Windows-based workstations, using a challenge-response mechanism to authenticate PPTP connections without sending passwords.

Extensible Authentication Protocol Transport Layer Security (EAP-TLS) is an Internet Engineering Task Force (IETF) open standard that uses the Transport Layer Security (TLS) protocol in Wi-Fi networks and PPP connections.

Public key infrastructure (PKI) is a set of roles, policies, and procedures needed to create, manage, distribute, use, store, and revoke digital certificates and manage public key encryption.

2.3.4.2 Layer 2 Tunneling Protocol

Layer 2 Tunneling Protocol (L2TP) is supported by most operating systems (including mobile devices). Although it provides no encryption by itself, it is considered secure when used together with IPsec (discussed in Section 2.3.4.6).

2.3.4.3 Secure Socket Tunneling Protocol

Secure Socket Tunneling Protocol (SSTP) is a VPN tunnel created by Microsoft to transport PPP or L2TP traffic through an SSL 3.0 channel. SSTP is primarily used for secure remote client VPN access, rather than for site-to-site VPN tunnels.

2.3.4.4 Microsoft Point-to-Point Encryption

Microsoft Point-to-Point Encryption (MPPE) encrypts data in PPP-based dial-up connections or PPTP VPN connections. MPPE uses the RSA RC4 encryption algorithm to provide data confidentiality and supports 40-bit and 128-bit session keys.

2.3.4.5 OpenVPN

OpenVPN is a highly secure, open-source VPN implementation that uses SSL/TLS encryption for key exchange. OpenVPN uses up to 256-bit encryption and can run over TCP or UDP. Although it is not natively supported by most major operating systems, it has been ported to most major operating systems, including mobile device operating systems.

2.3.4.6 Internet Protocol Security

IPsec is a secure communications protocol that authenticates and encrypts IP packets in a communication session. An IPsec VPN requires compatible VPN client software to be installed on the endpoint device. A group password or key is required for configuration. Client-server IPsec VPNs typically require user action to initiate the connection, such as launching the client software and logging in with a username and password.

A security association (SA) in IPsec defines how two or more entities will securely communicate over the network using IPsec. A single Internet Key Exchange (IKE) SA is established between communicating entities to initiate the IPsec VPN tunnel. Separate IPsec SAs are then established for each communication direction in a VPN session.

An IPsec VPN can be configured to force all of the user's internet traffic back through an organization's firewall, thus providing optimal protection with enterprise-grade security but with some performance loss. Alternatively, split tunneling can be configured to allow internet traffic from the device to go directly to the internet, while other specific types of traffic route through the IPsec tunnel, for acceptable protection with much less performance degradation.

If split tunneling is used, a personal firewall should be configured and active on the organization's endpoints because a split tunneling configuration can create a "side door" into the organization's network. Attackers can essentially bridge themselves over the internet, through the client endpoint, and into the network over the IPsec tunnel.

2.3.4.7 Secure Sockets Layer

Secure Sockets Layer (SSL) is an asymmetric encryption protocol used to secure communication sessions. SSL has been superseded by *Transport Layer Security* (TLS), although SSL is still the more commonly used terminology.

Key Terms

Secure Sockets Layer (SSL) is a cryptographic protocol for managing authentication and encrypted communication between a client and a server to protect the confidentiality and integrity of data exchanged in the session.

Transport Layer Security (TLS) is the successor to SSL (although it is still commonly referred to as SSL).

An SSL VPN can be deployed as an agent-based or agentless browser-based connection. An agentless SSL VPN requires users only to launch a web browser, open a VPN portal or webpage using the HTTPS protocol, and log in to the network with their user credentials. An agent-based SSL client is used within the browser session, which persists only while the connection is active and removes itself when the connection is closed. This type of VPN can be particularly useful for remote users who are connecting from an endpoint device they do not own or control, such as a hotel kiosk, where full client VPN software cannot be installed.

SSL VPN technology has become the de facto standard and preferred method of connecting remote endpoint devices back to the enterprise network, and IPsec is most commonly used in site-to-site or device-to-device VPN connections, such as connecting a branch office network to a headquarters location network or data center.

2.3.5 Data loss prevention

Network *data loss prevention* (DLP) solutions inspect data that is leaving, or egressing, a network (for example, via email, file transfer, or internet uploads, or by copying to a USB thumb drive) and prevent certain sensitive data – based on defined policies – from leaving the network. Sensitive data may include:

- Personally identifiable information (PII) such as names, addresses, birthdates, Social Security numbers, health records (including *electronic medical records,* or EMRs, and *electronic health records*, or EHRs), and financial data (such as bank account numbers and credit card numbers)
- Classified materials (such as military or national security information)

- Intellectual property, trade secrets, and other confidential or proprietary company information

Key Terms

As defined by HealthIT.gov, an *electronic medical record* (EMR) "contains the standard medical and clinical data gathered in one provider's office."

As defined by HealthIT.gov, an *electronic health record* (EHR) "go[es] beyond the data collected in the provider's office and include[s] a more comprehensive patient history. EHR data can be created, managed, and consulted by authorized providers and staff from across more than one healthcare organization."

A DLP security solution prevents sensitive data from being transmitted outside the network by a user, either inadvertently or maliciously. A robust DLP solution can detect the presence of certain data patterns even if the data is encrypted.

However, these solutions introduce a potential new vulnerability in the network because they have visibility into – and the ability to decrypt – all data on the network. Other methods rely on decryption happening elsewhere, such as on a web security appliance or other man-in-the-middle decryption engine. DLP solutions also often require many moving parts to effectively route traffic to and from inspection engines, which can add to the complexity of troubleshooting network issues.

2.3.6 Unified threat management

Unified threat management (UTM) devices combine numerous security functions into a single appliance, including:

- Anti-malware
- Anti-spam
- Content filtering
- DLP
- Firewall (stateful inspection)
- IDS/IPS
- VPN

UTM devices don’t necessarily perform any of these security functions better than their standalone counterparts, but they nonetheless serve a purpose in small to medium-size enterprise networks as a convenient and inexpensive solution that gives an organization an all-in-one security device. Typical disadvantages of UTM include:

- In some cases, they have reduced feature sets to make them more affordable.
- All security functions use the same processor and memory resources. Enablement of all the functions of a UTM can result in up to a 97 percent drop in throughput and performance, as compared to top-end throughput without security features enabled.
- Despite numerous security functions running on the same platform, the individual engines operate in silos with little or no integration or cooperation between them.

2.3 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **True or False.** A dynamic packet filtering firewall inspects each individual packet during a session to determine if the traffic should be allowed, blocked, or dropped by the firewall.

2. **Multiple Choice.** What are three characteristics of an application firewall? (Choose three.)

    a) proxies traffic rather than permitting direct communication between hosts

    b) can be used to implement strong user authentication

    c) masks the internal network from untrusted networks

    d) is extremely fast and has no impact on network performance

3. **Multiple Choice.** Which VPN technology is currently considered the preferred method for securely connecting a remote endpoint device back to an enterprise network? (Choose one.)

    a) Point-to-Point Tunneling Protocol (PPTP)

    b) Secure Socket Tunneling Protocol (SSTP)

    c) Secure Sockets Layer (SSL)

    d) Internet Protocol Security (IPsec)

4. **Multiple Choice.** Which is *not* a characteristic of unified threat management (UTM)? (Choose one.)

    a) It combines security functions such as firewalls, intrusion detection systems (IDSs), anti-malware, and data loss prevention (DLP) in a single appliance.

    b) Enabling all of the security functions in a UTM device can have a significant performance impact.

    c) It fully integrates all the security functions installed on the device.

    d) It can be a convenient solution for small networks.

2.4 Endpoint security

Traditional endpoint security encompasses numerous security tools, such as anti-malware software, anti-spyware software, personal firewalls, host-based intrusion prevention systems (HIPSs), and mobile device management (MDM) software. Endpoint security also requires implementation of effective endpoint security best practices, including patch management and configuration management.

2.4.1 Endpoint security basics

Endpoint security begins with a standard ("golden") image that ensures consistent configuration of devices across the organization, including:

- Disabling or removing operating system features and services that are not needed ("hardening")
- Installing current security updates
- Installing core applications

In practice, an organization will deploy numerous golden images, to, for example, support different device types, workgroups or departments, and user types (such as standard users and power users).

Most organizations deploy several security products to protect their endpoints, including personal firewalls, host-based intrusion prevention systems (HIPSs), mobile device management (MDM), mobile application management (MAM), DLP, and antivirus software. Nevertheless, cyber breaches continue to increase in frequency, variety, and sophistication. Additionally, the numbers and types of endpoints – including mobile and IoT devices – have grown exponentially and increased the attack surface. New variants of the Gafgyt, Mirai, and Muhstik botnets, among others, specifically target IoT devices. Additionally, new search engines, such as Shodan (Shodan.io), can automate the search for vulnerable internet-connected endpoints. Faced with the rapidly changing threat landscape, traditional endpoint security solutions and antivirus can no longer prevent security breaches on the endpoint.

Endpoint security is an essential element of cybersecurity because the network firewall cannot completely protect hosts from zero-day exploits. Zero-day exploits target unknown vulnerabilities in operating system and application software on host machines. Network firewalls may not be able to block an attacker's delivery of a zero-day exploit until a new signature identifying the zero-day attack has been developed and delivered to the firewall.

Network firewalls also may be restricted from decrypting all traffic because of regulations and laws. This restriction provides a window of opportunity for attackers to bypass a firewall's protection and exploit a host machine, necessitating endpoint security protection. Endpoint security protection is provided by an application that runs on the host machine. Effective endpoint security must be able to stop malware, exploits, and ransomware before they can compromise the host; provide protection while endpoints are online and offline; and detect threats and automate containment to minimize impact.

2.4.2 Malware protection

Malware protection – more specifically, antivirus software – has been one of the first and most basic tenets of information security since the early 1980s. Unfortunately, all of this hard-earned experience doesn't necessarily mean that the war is being won. For example, Trustwave's 2019 *Global Security Report* found that infection to detection of malware "in the wild" takes an average of 55 days.[35] Interestingly, web-based zero-day attacks, on average, remain "in the wild" up to four times longer than email-based threats because of factors that include user awareness of email-borne threats, availability and use of email security solutions (such as anti-spam and antivirus), and preferred use of the web as a threat vector by malware developers.

This poor "catch rate" is because of several factors. Some malware can mutate or can be updated to avoid detection by traditional anti-malware signatures. Also, advanced malware is increasingly specialized to the point where an attacker can develop customized malware that is targeted against a specific individual or organization.

Traditional anti-malware software uses various approaches to detect and respond to malware threats, including signature-based, container-based, application whitelisting, and anomaly-based techniques.

Note

With the proliferation of advanced malware such as remote access Trojans (RATs), anti-AV, and rootkits/bootkits (discussed in Section 1.2.1), security vendors have largely rebranded their antivirus solutions as "anti-malware" and expanded their malware protections to encompass the broader malware classifications.

[35] "2019 Trustwave Global Security Report." Trustwave. 2019. https://www.trustwave.com/en-us/resources/library/documents/2019-trustwave-global-security-report/.

2.4.2.1 Signature-based anti-malware software

Signature-based antivirus (or anti-malware) software is the oldest and most commonly used approach for detecting and identifying malware on endpoints. This approach requires security vendors to continuously collect malware samples, create matching signature files for those samples, and distribute those signature files as updates for their endpoint security products to all of their customers.

Deployment of signature-based antivirus software requires installing an engine that typically has kernel-level access to an endpoint's system resources. Signature-based antivirus software scans an endpoint's hard drive and memory, based on a predefined schedule and in real time when a file is accessed. If a known malware signature is detected, the software performs a predefined action, such as:

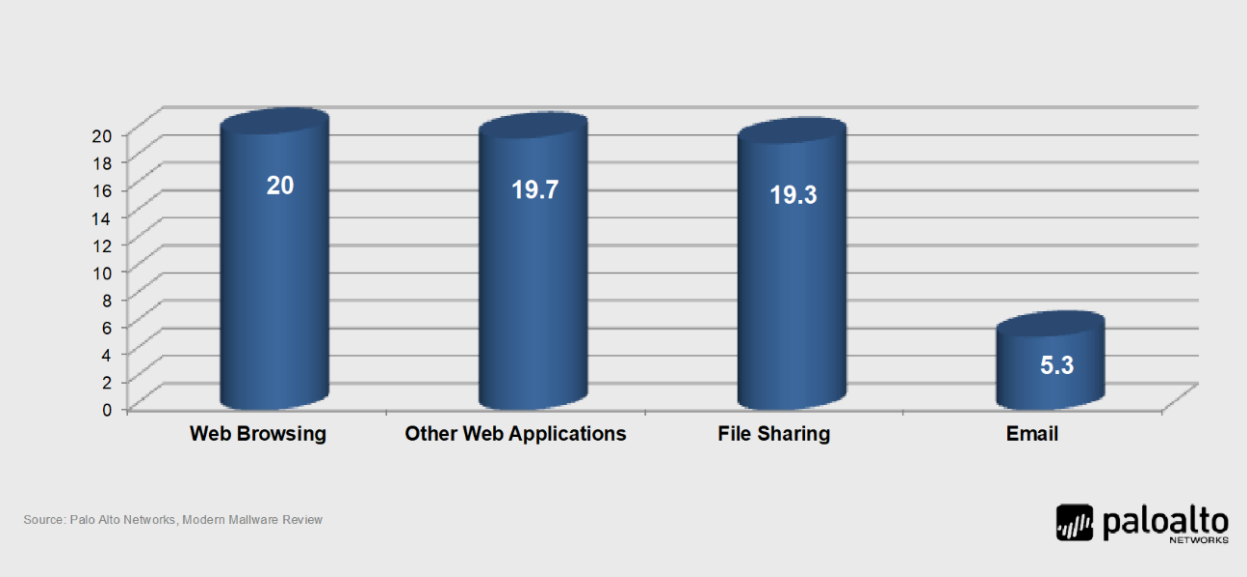
- **Quarantine.** Isolates the infected file so that it cannot infect the endpoint or other files
- **Delete.** Removes the infected file
- **Alert.** Notifies the user (and/or system administrator) that malware has been detected

Updated signatures must be regularly and frequently downloaded from the security vendor and installed on the organization's endpoints. Downloading and processing signature files in this manner can cause noticeable performance degradations on the networks and endpoints on which they are running.

Although the signature-based approach is very popular, its effectiveness is limited. By design, it is a reactive countermeasure because a signature file for new malware can't be created and delivered until the malware is already "in the wild," during which time networks and endpoints are blind to the threat – the notorious zero-day threat (or attack). The "zero-day" label is misleading, however, because the number of days from release to detection averages 5 to 20 days (see Figure 2-3).

Figure 2-3

Average time to detection by application vector



A sample of new or unknown suspicious traffic must first be captured and identified before a detection signature can be created by security vendors. The new signature must then be downloaded and installed on an organization's endpoints to provide protection.

This process means that some users and networks will be successfully breached by new malware until a new detection signature is created, downloaded, and installed. This reactive model creates a window of opportunity for attackers, leaving endpoints vulnerable – sometimes for weeks or even months – until new malware is suspected, collected, analyzed, and identified. During this time, attackers can infect networks and endpoints.

Another challenge for the signature-based approach is that millions of new malware variations are created each year (on average about 20,000 new forms daily), for which unique signatures must be written, tested, and deployed – after the new malware variation is discovered and sampled. Despite the fact that 70 percent of these millions of malware variations are based on a relatively limited number of malware "families" – numbering just seven in 2005 and increasing to only 20 over the past decade[36] – this reactive approach is not effective for protecting endpoints against modern malware threats.

Also, advanced malware uses techniques such as metamorphism and polymorphism to take advantage of the inherent weaknesses of signature-based detection to avoid being discovered in the wild and to circumvent signatures that have already been created. These techniques are

[36] Ibid.

so commonly used that "70 to 90 percent of malware samples [collected] today are unique to a single organization."[37]

2.4.2.2 Container-based endpoint protection

Container-based endpoint protection wraps a protective virtual barrier around vulnerable processes while they're running. If a process is malicious, the container detects it and shuts it down, preventing it from damaging other legitimate processes or files on the endpoint.

However, the container-based approach typically requires a significant amount of computing resource overhead, and attacks have been demonstrated that circumvent or disable container-based protection. This approach also requires knowledge of the applications that need to be protected and how they interact with other software components. As a result, a containerization tool will be developed to support certain common applications but will not be capable of protecting most proprietary or industry-specific software. Even web browser plugins and the like can have problems operating correctly within a container-based environment.

2.4.2.3 Application whitelisting

Application whitelisting is another endpoint protection technique that is commonly used to prevent end users from running unauthorized applications – including malware – on their endpoints.

Application whitelisting requires a positive control model in which no applications are permitted to run on the endpoint unless they're explicitly permitted by the whitelist policy. In practice, application whitelisting requires a large administrative effort to establish and maintain a list of approved applications. This approach is based on the premise that if you create a list of applications that are specifically allowed and then prevent any other file from executing, you can protect the endpoint. Although this basic functionality can be useful to reduce the attack surface, it is not a comprehensive approach to endpoint security.

Modern trends such as cloud and mobile computing, consumerization, and bring your own device (BYOD) and bring your own access (BYOA) make application whitelisting extremely difficult to enforce in the enterprise. Also, after an application is whitelisted, it is permitted to run even if the application has a vulnerability that can be exploited. An attacker can then simply exploit a whitelisted application and gain complete control of the target endpoint regardless of the whitelisting. After the application has been successfully exploited, the attacker can run malicious code while keeping all of the activity in memory. Since no new files are created and

[37] Ibid.

no new executables attempt to run, whitelisting software is rendered ineffective against this type of attack.

2.4.2.4 Anomaly detection

Endpoint security approaches that use mathematical algorithms to detect unusual activity on an endpoint are known as heuristics-based, behavior-based, or anomaly-detection solutions. This approach relies on first establishing an accurate baseline of what is considered "normal" activity. This approach has been around for many years and requires a very large dataset to reduce the number of false positives.

Key Terms

In anti-malware, a *false positive* incorrectly identifies a legitimate file or application as malware. A *false negative* incorrectly identifies malware as a legitimate file or application. In intrusion detection, a false positive incorrectly identifies legitimate traffic as a threat, and a false negative incorrectly identifies a threat as legitimate traffic.

2.4.3 Anti-spyware software

Anti-spyware software is very similar to traditional antivirus software because it uses signatures to look for other forms of malware beyond viruses, such as adware, malicious web application components, and other malicious tools, which share user behaviors without the user's permission.

2.4.4 Personal firewalls

Network firewalls protect an enterprise network against threats from an external network, such as the internet. However, most traditional port-based network firewalls do little to protect endpoints inside the enterprise network from threats that originate from within the network, such as another device that has been compromised by malware and is propagating throughout the network.

Personal (or host-based) firewalls are commonly installed and configured on laptop and desktop PCs. Personal firewalls typically operate as Layer 7 (Application layer) firewalls that allow or block traffic based on an individual (or group) security policy. Personal firewalls are particularly helpful on laptops used by remote or traveling users who connect their laptop computers directly to the internet (for example, over a public Wi-Fi connection). Also, a personal firewall can control outbound traffic from the endpoint to help prevent the spread of malware from that endpoint. However, note that disabling or otherwise bypassing a personal firewall is a common and basic objective in most advanced malware today.

Windows Firewall is an example of a personal firewall that is installed as part of the Windows desktop or mobile operating system. A personal firewall protects only the endpoint device that it is installed on, but it provides an extra layer of protection inside the network.

2.4.5 Host-based intrusion prevention systems

HIPS is another approach to endpoint protection that relies on an agent installed on the endpoint to detect malware. A HIPS can be either signature-based or anomaly-based, and is therefore susceptible to the same issues as other signature and anomaly-based endpoint protection approaches.

Also, HIPS software often causes significant performance degradation on endpoints. A recent Palo Alto Networks survey found that 25 percent of respondents indicated HIPS solutions "caused significant end user performance impact."

2.4.6 Mobile device management

Mobile device management (MDM) software provides endpoint security for mobile devices such as smartphones and tablets. Centralized management capabilities for mobile devices provided by MDM include:

- **Data loss prevention (DLP).** Restrict what type of data can be stored on or transmitted from the device.
- **Policy enforcement.** Require passcodes, enable encryption, lock down security settings, and prevent *jailbreaking* or *rooting*, for example.
- **Malware protection.** Detect and prevent mobile malware.
- **Software distribution.** Remotely install software, including patches and updates over a cellular or Wi-Fi network.
- **Remote erase/wipe.** Securely and remotely delete the complete contents of a lost or stolen device.
- **Geofencing and location services.** Restrict specific functionality in the device based on its physical location.

Key Terms

Jailbreaking refers to hacking an Apple iOS device to gain root-level access to the device. Jailbreaking is sometimes done by end users to allow them to download and install mobile apps without paying for them, from sources other than the App Store that are not sanctioned and/or controlled by Apple. Jailbreaking bypasses the security features of the device by replacing the firmware's operating system with a similar, albeit counterfeit version, which makes it vulnerable to malware and exploits. Jailbreaking is known as *rooting* on Google Android devices.

2.4 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **True or False.** Signature-based anti-malware software is considered a proactive security countermeasure.
2. **Fill in the Blank.** _____ endpoint protection wraps a protective virtual barrier around vulnerable processes while they're running.
3. **Short Answer.** What is the main disadvantage of application whitelisting related to exploit prevention?
4. **Multiple Choice.** What are three typical mobile device management software capabilities? (Choose three.)
    a) data loss prevention (DLP)
    b) policy enforcement
    c) intrusion detection
    d) malware prevention

2.5 Server and system administration

Server and system administrators perform a variety of important tasks in a network environment. Typical server and system administration tasks include:

- Account provisioning and deprovisioning
- Managing account permissions
- Installing and maintaining server software
- Maintaining and optimizing servers, applications, databases (may be assigned to a database administrator), network devices (may be assigned to a network administrator), and security devices (may be assigned to a security administrator)
- Installing security patches
- Managing system and data backup and recovery
- Monitoring network communication and server logs
- Troubleshooting and resolving server and system issues

2.5.1 Identity and access management

Identity and access management (IAM) provides authentication, authorization, and access control functions. IAM tools provide control for the provisioning, maintenance, and operation of identities — including users, devices, and services — and the level of access to network, data center, and cloud resources that different identities are permitted. Authentication is the process of verifying an identity based on one of (single-factor authentication) or a combination of (multi-factor authentication) the following three factors:

- **Something you know** (such as a password or PIN)
- **Something you have** (such as a security token or authenticator app)
- **Something you are** (biometrics, such as a fingerprint or retina pattern)

Multi-factor authentication (MFA) is becoming more common as organizations recognize the inherent weaknesses associated with single-factor authentication techniques such as passwords and PINs. MFA typically requires a username (or user ID), password, and a one-time passcode sent to a smartphone via SMS text, authenticator app (such as Microsoft Authenticator or Google Authenticator), or a hardware token. The one-time passcode is only valid for a limited period of time (typically one to five minutes) and only for a single login session. Many organizations implement MFA with dynamic policies to limit the number of times users are prompted for MFA. For example, the organization might require MFA only once every five days or only if the user is authenticating from a different IP address than usual.

Effective IAM requires organizations to implement well-defined processes for account provisioning/deprovisioning, role-based access control (RBAC), privilege management, and access reviews to ensure user accounts (and other directory objects) are promptly created and disabled/deleted, least privilege access is enforced, and good user/network hygiene is maintained.

2.5.2 Directory services

A directory service is a database that contains information about users, resources, and services in a network. The directory service associates users and network permissions to control who has access to which resources and services on the network. Directory services include:

- **Active Directory.** A centralized directory service developed by Microsoft for Windows networks to provide authentication and authorization of users and network resources. Active Directory uses Lightweight Directory Access Protocol (LDAP), *Kerberos*, and the Domain Name System (DNS, discussed in Section 2.0.4).
- **Lightweight Directory Access Protocol (LDAP).** An IP-based client-server protocol that provides access and manages directory information in TCP/IP networks.

Key Terms

Kerberos is an authentication protocol in which tickets are used to identify network users.

2.5.3 Vulnerability and patch management

New software vulnerabilities and exploits are discovered all the time, requiring diligent software patch management by system and security administrators in every organization.

However, patch management protects an organization's endpoints only after a vulnerability has been discovered and the patch installed. Delays of days, weeks, or longer are inevitable because security patches for newly discovered vulnerabilities must be developed, distributed, tested, and deployed. Although patch management is an important aspect of any information security program, like signature-based anti-malware detection, it is an endless race against time that offers no protection against zero-day exploits.

Organizations should also proactively perform regular vulnerability assessments to identify, evaluate, quantify, and prioritize security weaknesses in their applications and systems. Vulnerability assessments may consist of port scans, vulnerability scans, and/or penetration tests.

2.5.4 Configuration management

Configuration management is the formal process used by organizations to define and maintain standard configurations for applications, devices, and systems throughout their lifecycle. For example, a particular desktop PC model may be configured by an organization with specific security settings, such as enabling whole disk encryption and disabling USB ports. Within the desktop operating system, security settings such as disabling unneeded and risky services (for example, FTP and Telnet) may be configured. Maintenance of standard configurations on applications, devices, and systems used by an organization helps reduce risk exposure and improve security posture.

2.5.5 Structured host and network troubleshooting

A network or segment of a network that goes down could have a negative impact on your organization or business. Network administrators should use a systematic process to troubleshoot network problems when they occur to restore the network to full production as quickly as possible without causing new issues or introducing new security vulnerabilities. The troubleshooting process performed by a network administrator to resolve network problems quickly and efficiently is a skill that is highly sought after in IT.

Two of the most important troubleshooting tasks a network administrator performs occur long before a network problem occurs: baselining and documenting the network.

A baseline provides quantifiable metrics that are periodically measured with various network performance monitoring tools, protocol analyzers, and packet sniffers. Important metrics might include application response times, server memory and processor utilization, average and peak network throughput, and storage input/output operations per second. These baseline metrics provide an important snapshot of normal network operations to help network administrators identify impending problems, troubleshoot current problems, and know when a problem has been fully resolved.

Network documentation should include logical and physical diagrams, application data flows, change management logs, user and administration manuals, and warranty and support information. Network baselines and documentation should be updated any time a significant change to the network occurs and as part of the change management process of an organization.

Many formal multi-step troubleshooting methodologies have been published, and organizations or individual network administrators may have their own preferred method. Generally speaking, however, troubleshooting consists of these steps:

1. Discover the problem.
2. Evaluate the system configuration against the baseline.
3. Track the possible solutions.
4. Execute a plan.
5. Check the results.
6. Verify the solution. (If unsuccessful, return to Step 2. If successful, go to Step 7.)
7. Deploy the positive solution.

Troubleshooting host and network connectivity problems typically starts with analyzing the scope of the problem and identifying the devices and services that are affected. Problems with local hosts are typically much easier to assess and remedy than problems that affect a network segment or service. For an individual device that loses network connectivity, the problem sometimes can be easily resolved by simply restarting the device. However, problems with integrated or shared services (for example, web or file services) can be complex, and restarting a service or rebooting a device may actually compound the problem. Connectivity problems may be intermittent or difficult to trace, so it is important that your troubleshooting processes follow an approved or standardized methodology.

The OSI model (discussed in Section 2.2.1) provides a logical model for troubleshooting complex host and network issues. Depending on the situation, you might use the bottom-up, top-down, or divide-and-conquer approach discussed in the following paragraphs when you use the OSI model to guide your troubleshooting efforts. In other situations, you might make an educated guess about the source of the issue and begin investigating at the corresponding layer of the OSI model, or use the substitution method (replacing a bad component with a known good component) to quickly identify and isolate the cause of the issue.

When you use a bottom-up approach to diagnose connectivity problems, you begin at the Physical layer of the OSI model by verifying network connections and device availability. For example, a wireless device may have power to the antenna or transceiver temporarily turned off. Or a wireless access point may have lost power because a circuit breaker was tripped offline or a fuse was blown. Similarly, a network cable connection may be loose, or the cable may be damaged. Thus, before you begin inspecting service architectures, you should start with the basics: confirm physical connectivity.

Moving up to the Data Link layer, you verify data link architectures, such as compatibility with a particular standard or frame type. Although Ethernet is a predominant LAN network standard,

devices that roam (such as wireless devices) sometimes automatically switch between Wi-Fi, Bluetooth, and Ethernet networks. Wireless networks usually have specified encryption standards and keys. Connectivity may be lost because a network device or service has been restored to a previous setting, and the device is not responding to endpoint requests that are using different settings. Firewalls and other security policies may also be interfering with connection requests. You should never disable firewalls, but in a controlled network environment with proper procedures established, you may find that temporarily disabling or bypassing a security appliance resolves a connectivity issue. The remedy then is to properly configure security services to allow the required connections.

Various connectivity problems may also occur at the Network layer. Important troubleshooting steps include confirming proper network names and addresses. Devices may have improperly assigned IP addresses that are causing routing issues or IP address conflicts on the network. A device may have an improperly configured IP address because it cannot communicate with a DHCP server on the network. Similarly, networks have different identities, such as wireless SSIDs, domain names, and workgroup names. Another common problem exists when a particular network has conflicting names or addresses. Issues with DNS name resolvers may be caused by DNS caching services or connection to the wrong DNS servers. *Internet Control Message Protocol* (ICMP) is used for network control and diagnostics at the Network layer of the OSI model. Commonly used ICMP commands include **`ping`** and **`traceroute`**. These two simple but powerful commands (and other ICMP commands and options) are some of the most commonly used tools for troubleshooting network connectivity issues. You can run ICMP commands in the command-line interface on computers, servers, routers, switches, and many other networked devices.

Key Terms

Internet Control Message Protocol (ICMP) is an internet protocol used to transmit diagnostic messages.

At the Transport layer, communications are more complex. Latency and network congestion can interfere with communications that depend on timely acknowledgments and handshakes. Time-to-live (TTL) values sometimes must be extended in the network service architecture to allow for slower response times during peak network traffic hours. Similar congestion problems can occur when new services are added to an existing network or when a local device triggers a prioritized service, such as a backup or an antivirus scan.

Session layer settings can also be responsible for dropped network connections. For example, devices that automatically go into a power standby mode (“sleep”) may have expired session

tokens that fail when the device attempts to resume connectivity. At the server, failover communications or handshake negotiations with one server may not translate to other clustered servers. Sessions may have to be restarted.

Presentation layer conflicts are often related to changes in encryption keys or updates to service architectures that are not supported by various client devices. For example, an older browser may not interoperate with a script or a new encoding standard.

Application layer network connectivity problems are extremely common. Many applications may conflict with other apps. Apps also may have caching or corrupted files that can be remedied only by uninstalling and reinstalling or by updating to a newer version. Some apps also require persistent connections to updater services or third parties, and network security settings may prevent those connections from being made.

Other troubleshooting steps may include searching log files for anomalies and significant events, verifying that certificates or proper authentication protocols are installed and available, verifying encryption settings, clearing application caches, updating applications, and, for endpoints, removing and reinstalling an application. Search vendor-supported support sites and forums and frequently asked questions (FAQ) pages before you make changes to installed services. You also must be aware of any service-level agreements (SLAs) that your organization is required to meet.

Always follow proper troubleshooting steps, keep accurate records of any changes that you attempt, document your changes, and publish any remedies so that others can learn from your troubleshooting activities.

2.5 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Fill in the Blank.** _____ tools provide control for the provisioning, maintenance, and operation of user identities and the level of access to network, data center, and cloud resources that different identities are permitted.

2. **Fill in the Blank.** The _____ provides a logical model for troubleshooting complex host and network issues.

2.6 Secure the Enterprise (Strata)

The networking infrastructure of an enterprise can be extraordinarily complex. The Security Operating Platform secures enterprise networks' perimeters, data centers, and branches with a fully integrated and automated platform that simplifies security. Simplifying your security posture allows you to reduce operational costs and the supporting infrastructure while increasing your ability to prevent threats to your organization and quickly adjust to your dynamic environment. The key Security Operating Platform elements for securing the enterprise are:

- **Next-generation firewall.** The foundation of the Security Operating Platform available in physical, virtual, and cloud-delivered deployment options to provide consistent protection wherever your data and apps reside
- **Subscription services.** Add-on enhanced threat services and next-generation firewall capabilities, including DNS Security, URL Filtering, Threat Prevention, WildFire malware prevention and others
- **Panorama.** Provides centralized network security management, simplifying administration while delivering comprehensive controls and deep visibility into network-wide traffic and security threats
- **Okyo Garde.** Transforms the employee home network into a trusted enterprise edge by leveraging Prisma Access to bring Secure Access Service Edge (SASE) to employee home networks with unified corporate security policy management to enable secure work-from-home (WFH) models

2.6.1 Next-generation firewall

Fundamental shifts in application usage, user behavior, and complex network infrastructure have created a threat landscape that exposes weaknesses in traditional port-based network firewalls. End users want access to an ever-increasing number of applications, operating across a wide range of device types, often with little regard for the business or security risks. Meanwhile, data center expansion, network segmentation, virtualization, and mobility initiatives are forcing organizations to rethink how to enable access to applications and data, while protecting their networks from a new, more sophisticated class of advanced threats that evade traditional security mechanisms.

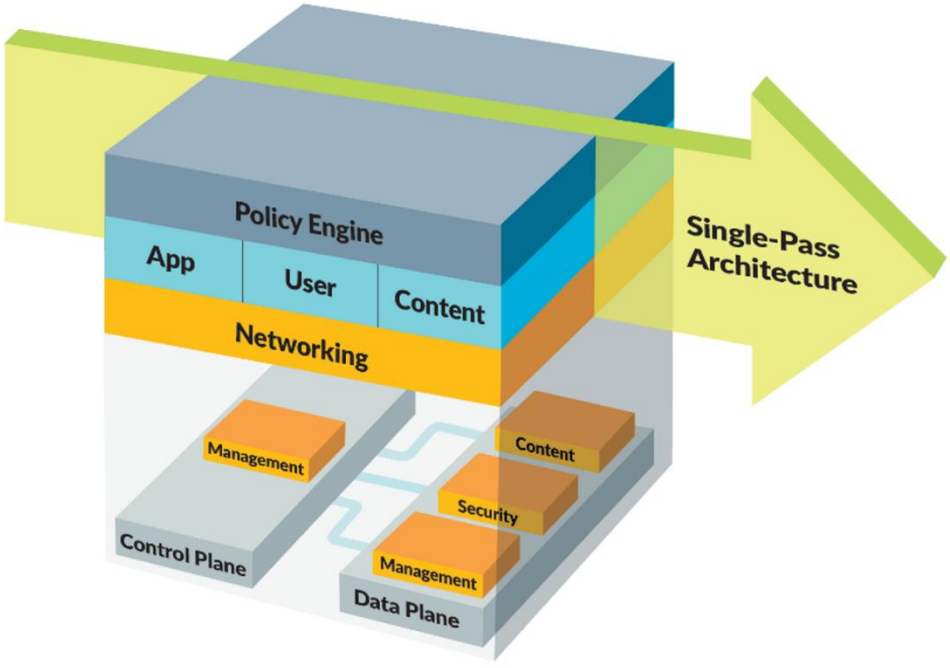
Palo Alto Networks next-generation firewalls are the core of the Security Operating Platform. The next-generation firewall inspects all traffic – including applications, threats, and content –

and associates it with the user, regardless of location or device type. The application, content, and user become integral components of the enterprise security policy.

Palo Alto Networks next-generation firewalls are built on a single-pass architecture (see Figure 2-4), which is a unique integration of software and hardware that simplifies management, streamlines processing, and maximizes performance. The single-pass architecture integrates multiple threat prevention disciplines (IPS, anti-malware, URL Filtering, etc.) into a single stream-based engine with a uniform signature format. This architecture allows traffic to be fully analyzed in a single pass without the performance degradation seen in multifunction gateways. The software is tied directly to a parallel processing hardware platform that uses function-specific processors for threat prevention, to maximize throughput, and to minimize latency.

Figure 2-4

Palo Alto Networks next-generation firewalls use a single-pass architecture.

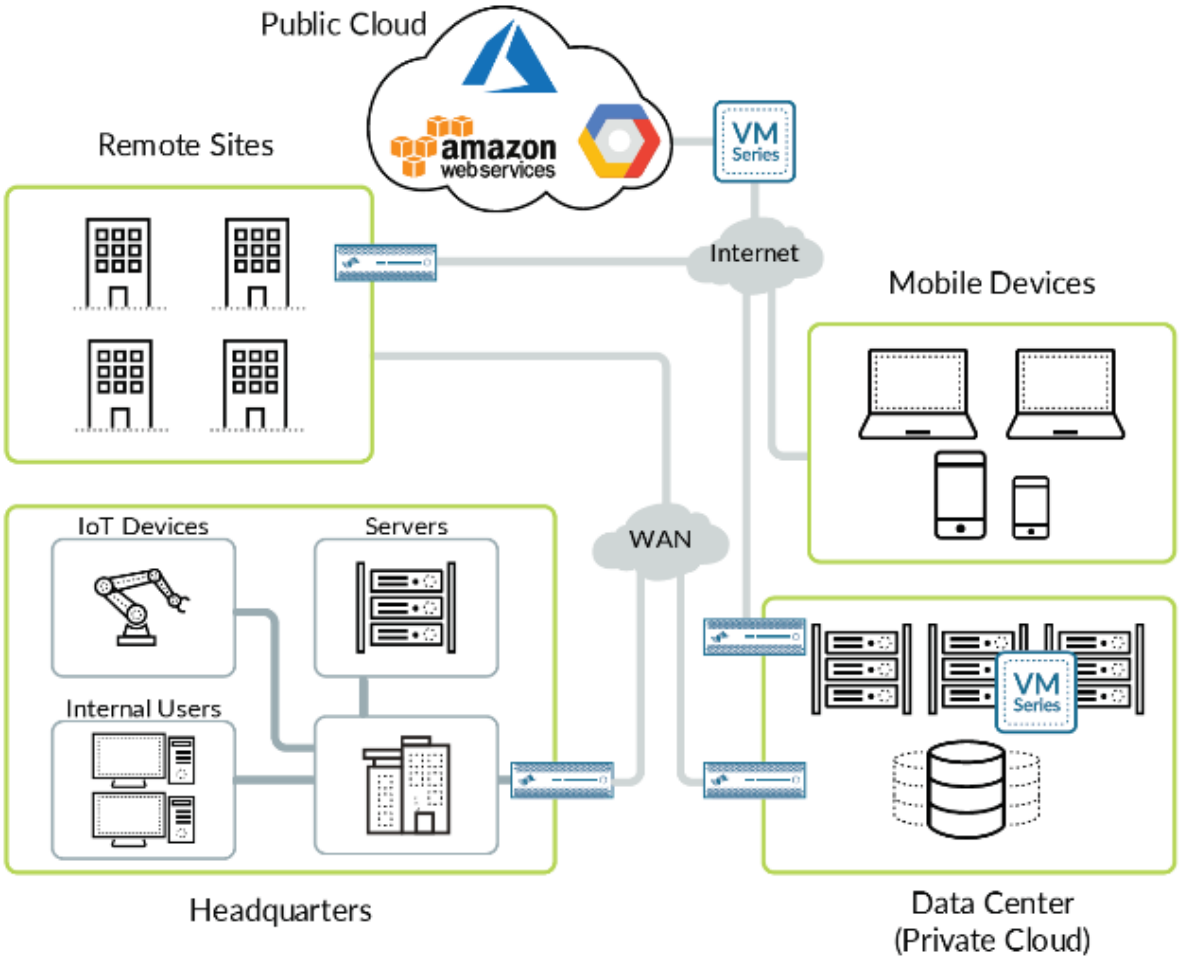


The use of one common engine means that two key benefits are realized. First, unlike file proxies that need to download the entire file before they can scan the traffic, a stream-based engine scans traffic in real time, only reassembling packets as needed and only in very small amounts. Second, unlike with traditional approaches, all traffic can be scanned with a single engine, instead of multiple scanning engines.

Organizations deploy next-generation firewalls at the network perimeter and inside the network at logical trust boundaries. All traffic crossing the next-generation firewall undergoes a full-stack, single-pass inspection, providing the complete context of the application, associated content, and user identity. With this level of context, you can align security with your key business initiatives (see Figure 2-5).

Figure 2-5

Next-generation firewall locations in the enterprise network



The next-generation firewall functions as a segmentation gateway in a Zero Trust architecture (discussed in Section 1.3.2). By creating a micro-perimeter, the next-generation firewall ensures that only known, allowed traffic or legitimate applications have access to the protect surface.

Next-generation firewalls include several key capabilities that enable complete visibility of the application traffic flows, associated content, and user identity and protect these three things from known, unknown, and advanced persistent threats. The essential functional capabilities in an effective next-generation firewall include:

- **Application identification.** Accurately identify applications regardless of port, protocol, evasive techniques, or encryption. Provide visibility of applications and granular policy-based control over applications, including individual application functions.
- **User identification.** Accurately identify users and subsequently use identity information as an attribute for policy control.
- **Content identification.** Content identification controls traffic based on complete analysis of all allowed traffic, using multiple threat prevention and data loss prevention techniques in a single-pass architecture that fully integrates all security functions.

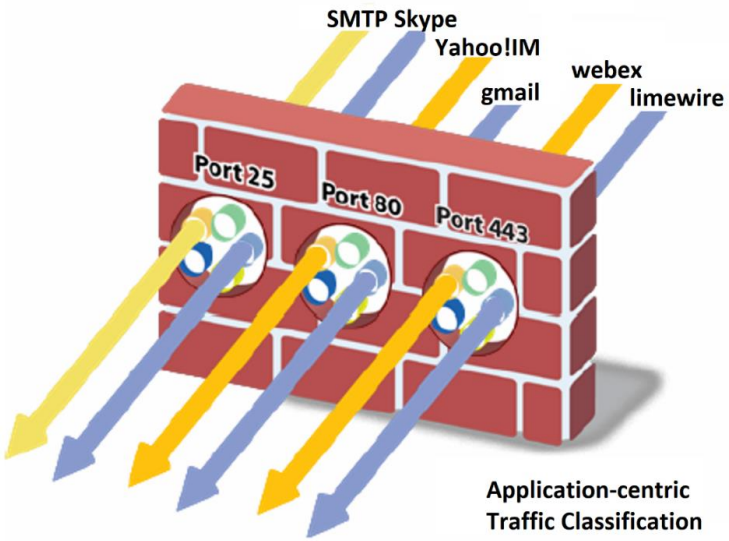
2.6.1.1 Application identification

Stateful packet inspection technology – the basis for most of today's legacy firewalls – was created more than 25 years ago, at a time when applications could be controlled using ports and source/destination IP addresses. The strict adherence to port-based classification and control methodology is the primary policy element; it is hard-coded into the foundation and cannot be turned off. As a result, many of today's applications cannot be identified, much less controlled, by the firewall, and no amount of "after the fact" traffic classification by firewall "helpers" can correct the firewall port-based classification.

Establishing port and protocol information is a first step in application identification, but it is insufficient by itself. Robust application identification and inspection in a next-generation firewall enables granular control of the flow of sessions through the firewall. Identification is based on the specific applications (such as Skype, Gmail, and WebEx) that are being used, instead of just relying on the underlying set of often indistinguishable network communication services (see Figure 2-6).

Figure 2-6

Application-centric traffic classification identifies specific applications on the network irrespective of the port and protocol in use.



Application identification provides visibility and control over work-related and non-work-related applications that can evade detection by legacy port-based firewalls, for example, by masquerading as legitimate traffic, hopping ports, or by using encryption to slip past the firewall.

Application identification (App-ID) technology in a Palo Alto Networks next-generation firewall does not rely on a single element, such as port or protocol. Instead, App-ID uses multiple mechanisms to determine, first and foremost, what the application is. The application identity then becomes the basis for the firewall policy that is applied to the session. App-ID is highly extensible, and, as applications continue to evolve, application detection mechanisms can be added or updated as a means of keeping pace with the ever-changing application landscape.

Many organizations are not fully aware of the number of applications in use, how heavily they are used, or by whom. This lack of visibility forces organizations to implement negative (blacklist) enforcement approaches where they selectively block traffic and destinations known to be a risk to the organization. The next-generation firewall also allows you to implement a positive (whitelist) enforcement policy where you selectively allow the applications required to run your organization. This significantly reduces the number of ways cybercriminals can attack your organization. A key to positive enforcement is App-ID. App-ID identifies the applications traversing the firewall – regardless of port or protocol – even if the traffic is tunneled in Generic Routing Encapsulation (GRE) tunnels, uses evasive tactics, or is encrypted. App-ID can determine the difference between base applications and application functions. This level of

visibility brings a complete understanding of the applications on your network and their value and risk to your organization.

App-ID traffic classification technology

The first task that a Palo Alto Networks next-generation firewall executes is using App-ID to identify the applications traversing the network. App-ID uses a multifaceted approach to determine the application, irrespective of port, protocol, encryption (SSL and SSH), or other evasive tactics employed. The number and order of identification mechanisms used to identify the application vary depending on the application. The application identification techniques (see Figure 2-7) used include:

- **Application signatures.** To identify an application, App-ID first uses signatures to look for unique application properties and related transaction characteristics. The signature also determines whether the application is using its default port or a non-standard port. Context-based signatures look for unique properties and transaction characteristics to correctly identify the application regardless of the port and protocol being used. These signatures include the ability to detect specific functions within applications (such as file transfers within SaaS applications). If the security policy allows the identified application, App-ID further analyzes the traffic in order to identify more granular applications and scan for threats.

- **TLS/SSL and SSH decryption.** If App-ID determines that TLS/SSL encryption is in use, it can decrypt and reevaluate the traffic. App-ID uses a similar approach with SSH in order to determine whether port forwarding is being used to tunnel traffic over SSH.

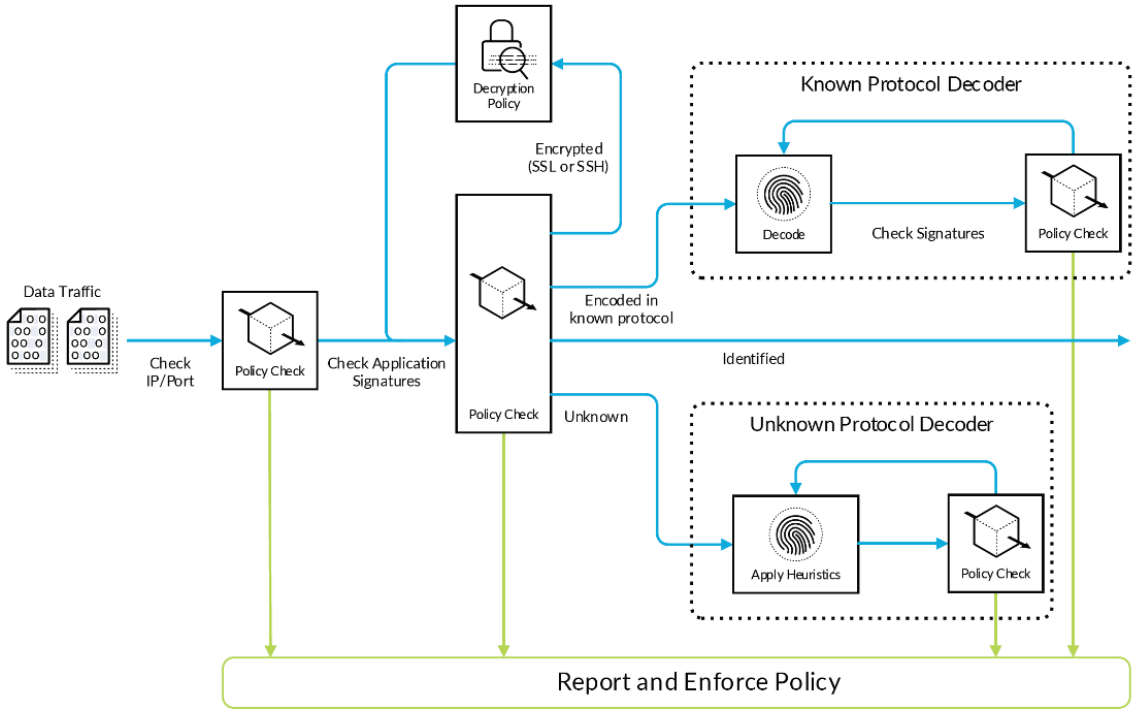
- **Application and protocol decoding.** For known protocols, decoders apply additional context-based signatures to detect applications tunneling inside the protocols. Decoders validate that traffic conforms to the protocol specification and support network address translation (NAT) traversal and opening dynamic pinholes for applications such as Voice over IP (VoIP) or File Transfer Protocol (FTP). Decoders for popular applications also identify the individual functions within the application. In addition to identifying applications, decoders identify files and other content to be scanned for threats or sensitive data.

- **Heuristics.** In certain cases, evasive applications cannot be detected by using advanced signature and protocol decoding. In those cases, App-ID uses heuristic or behavioral analysis to identify applications that use proprietary encryption, such as peer-to-peer (P2P) file sharing. Heuristic analysis, with the other App-ID techniques, provides visibility into applications that might otherwise elude identification. The heuristics are specific to

each application and include checks based on information such as the packet length, session rate, and packet source.

Figure 2-7

How Palo Alto Networks App-ID classifies applications



With App-ID as the foundational element for every Palo Alto Networks next-generation firewall, administrators can regain visibility into, and control over, the applications traversing the network.

App-ID: Addressing custom or unknown applications

Using the Application Command Center (ACC), you can see the applications in use across your organization. After you've determined the value of an application to your organization, App-ID controls the security policy for that application. The security policy can include a number of different actions, such as:

- Allowing or denying
- Allowing but scanning the content for exploits, viruses, and other threats
- Allowing based on schedule, users, or groups
- Controlling file or sensitive data transfer

- Allowing or denying a subset of application functions

While you compile the list of the applications you want to support, tolerate, or block, App-ID can restrict applications that behave in undesirable ways. You can use application categories, technologies, and risk ratings to define a security policy to block any applications that match those characteristics.

Often, safe application enablement means striking an appropriate security policy balance between allowing some application functions and denying others. Examples include:

- Allowing Facebook but denying Facebook mail, chat, posting, and apps, effectively allowing users only to browse Facebook.

- Allowing the use of SaaS applications such as Dropbox but denying file uploads. This technique grants internal users access to personal file shares but prevents intentional or unintended corporate information leaks.

The list of App-IDs is updated monthly, with new applications added based on input from the Palo Alto Networks community (customers and partners) and market trends. All App-IDs are classified by category, subcategory, technology, and risk rating. The security policy can use these classifications to automatically support new applications as the App-ID list expands. Alternatively, you can specify that you want to review new applications and determine how they are treated before the new list is installed.

Despite regular updates, unknown application traffic will inevitably still be detected on the network. This can include:

- **Unknown commercial applications.** Administrators can use the ACC and the log viewer to quickly determine whether an unknown application is a commercial application. Administrators can use the packet capture (pcap) feature on the Palo Alto Networks next-generation firewall, to record the traffic and submit it for App-ID development. The new App-ID is developed, tested with the organization, and then added to the global database for all users.

- **Internal or custom applications.** Administrators can use the ACC and the log viewer to quickly determine whether an unknown application is an internal or custom application. You can develop a custom App-ID for the application, using the exposed protocol decoders. The protocol decoders that have been exposed include:

    - FTP (File Transfer Protocol)

    - HTTP (Hypertext Transfer Protocol) and HTTPS (HTTP Secure, or HTTP over SSL)

- IMAP (Internet Message Access Protocol) and SMTP (Simple Mail Transfer Protocol)
- RTSP (Real Time Streaming Protocol)
- Telnet
- unknown-TCP, unknown-UDP, and file body (for html/pdf/flv/swf/riff/mov)

After the custom App-ID is developed, traffic identified by it is treated in the same manner as the previously classified traffic: it can be enabled via policy, inspected for threats, shaped using quality of service (QoS), etc. Alternatively, an application override can be created and applied, which effectively renames the application. Custom App-ID entries are managed in a separate database on the next-generation firewall to ensure that they are not impacted by weekly App-ID updates.

An important point to highlight is that Palo Alto Networks next-generation firewalls use a positive enforcement model, which means that all traffic can be denied except those applications that are expressly allowed via policy. This positive enforcement model means that in some cases the unknown traffic can be easily blocked or tightly controlled. Alternative offerings that are based on IPS will allow unknown traffic to pass through without providing any semblance of visibility or control.

App-ID in action: Identifying WebEx

When a user initiates a WebEx session, the initial connection is an SSL-based communication. With App-ID, the device sees the traffic and determines that it is using SSL. If there is a matching decryption policy rule, then the decryption engine and protocol decoders are initiated to decrypt the SSL and detect that it is HTTP traffic. After the decoder has the HTTP stream, App-ID can apply contextual signatures and detect that the application in use is WebEx.

WebEx is then displayed in the ACC and can be controlled via a security policy. If the end user initiates the WebEx Desktop Sharing feature, WebEx undergoes a “mode-shift:” the session is altered from a conferencing application to a remote access application. In this scenario, the characteristics of WebEx have changed, and App-ID detects the WebEx Desktop Sharing feature, which is then displayed in the ACC. At this stage, an administrator has learned more about the application use and can exert policy control over the use of the WebEx Desktop Sharing feature separately from general WebEx use.

Application identification and policy control

Application identification enables administrators to see the applications on the network, learn how they work, and analyze their behavioral characteristics and relative risk. When application identification is used in conjunction with user identification, administrators can see exactly who is using the application based on their identity instead of just an IP address. With this information, administrators can use granular rules – based on a positive security model – to block unknown applications while enabling, inspecting, and shaping those applications that are allowed.

After an application has been identified and a complete picture of its usage is gained, organizations can apply policies with a range of responses that are far more granular than the "allow" or "deny" actions available in legacy firewalls. Examples of this include:

- Allow or deny
- Allow but scan for exploits, viruses, and other threats
- Allow based on schedule, users, or groups
- Decrypt and inspect
- Apply traffic shaping through QoS
- Apply policy-based forwarding
- Allow certain application functions
- Any combination of the above

Application function control

For many organizations, secure application enablement means striking an appropriate security policy balance by enabling individual application functionality while blocking other functions within the same application. Examples may include:

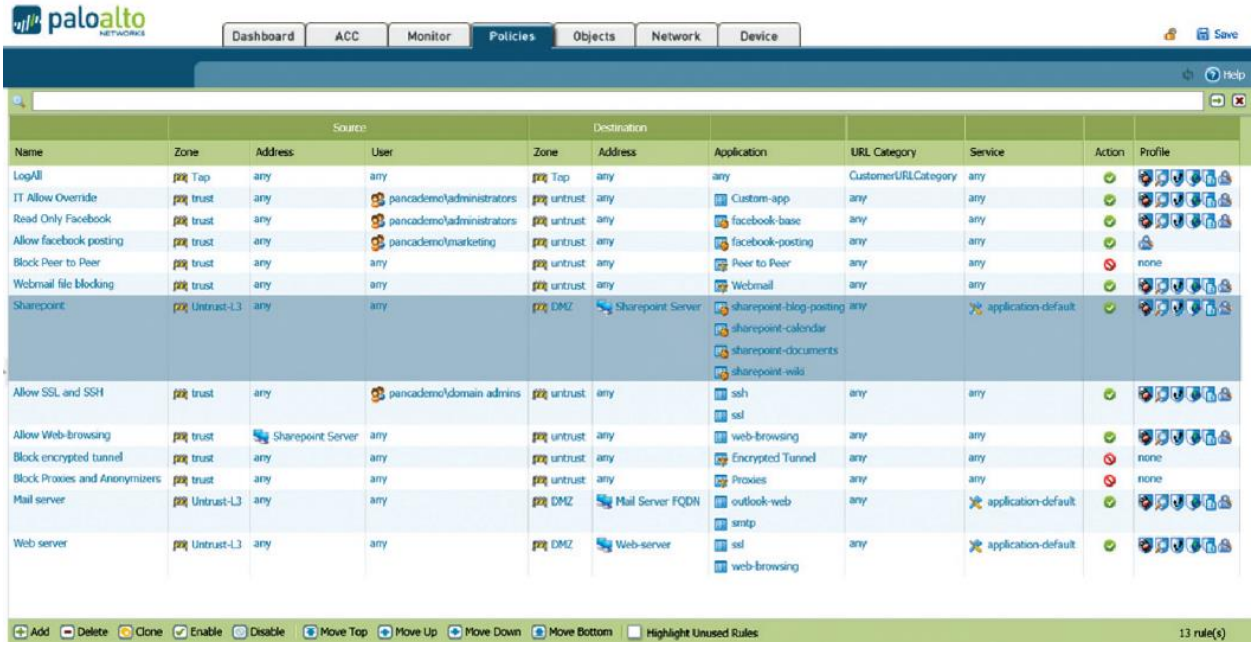
- Allowing SharePoint documents but blocking the use of SharePoint administration
- Blocking Facebook mail, chat, posting, and applications but allowing Facebook itself, effectively allowing users only to browse Facebook

App-ID uses an application hierarchy that follows a "container and supporting function" model to help administrators easily choose which applications to allow, while blocking or controlling

functions within the application. Figure 2-8 shows SharePoint as the container application and the individual functions within it.

Figure 2-8

Application function control maximizes productivity by safely enabling the application itself (Microsoft SharePoint) or individual functions.



Controlling multiple applications: Dynamic filters and groups

In some cases, organizations may want to control applications in bulk, as opposed to controlling them individually. The two mechanisms in the Palo Alto Networks next-generation firewall that address this need are application groups and dynamic filters:

- **Application groups.** A group of applications is a static list of applications that allows certain users access while blocking access for other users. For example, remote management applications such as Remote Desktop Protocol (RDP), Telnet, and Secure Shell (SSH) are commonly used by IT support personnel, yet employees who fall outside of these groups also use these tools to access their home networks. A group of applications can be created and assigned to IT support through User-ID (discussed in Section 2.6.1.2), binding the groups to the policy. New employees only need to be added to the directory group; no updates are needed to the policy itself.

- **Dynamic filters.** A dynamic filter is a set of applications that is created based on any combination of the filter criteria: category, subcategory, behavioral characteristic, underlying technology, or risk factor. After the desired filter is created, a policy that

blocks or enables and scans the traffic can be applied. As new App-ID files are added that fulfill the filter criteria, the filter is automatically updated as soon as the device is updated, thereby minimizing the administrative effort associated with policy management.

2.6.1.2 User identification

As you define security policies based on application use, a key component of that policy is who should be able to use those applications. IP addresses are ineffective identifiers of the user or the role of the server within the network. With the User-ID and dynamic address group (DAG) features, you can dynamically associate an IP address with a user or the role of a server in the data center. Afterward, you can define security policies that adapt dynamically to changing environments.

In environments that support multiple types of end users (for example, Marketing or Human Resources) across a variety of locations and access technologies, it is unrealistic to guarantee physical segmentation of each type of user. Visibility into the application activity at a user level, not just at an IP address level, allows you to enable the applications traversing the network more effectively. You can define both inbound and outbound policies to safely enable applications based on users or groups of users. Examples of user-based policies include:

- Enabling the IT department to use SSH, Telnet, and FTP on standard ports
- Allowing the Help Desk Services group to use Slack
- Allowing all users to read Facebook but blocking the use of Facebook apps and restricting posting to only employees in Marketing

User-ID: Integrating user information and security policies

Creating and managing security policies on a next-generation firewall, based on the application and the identity of the user regardless of device or location, is a more effective means of protecting the network than relying solely on port and IP address information in legacy, port-based firewalls. User-ID enables organizations to leverage user information stored in a wide range of repositories for the following purposes:

- **Visibility.** Improved visibility into application usage based on user and group information can help organizations maintain a more accurate picture of network activity.

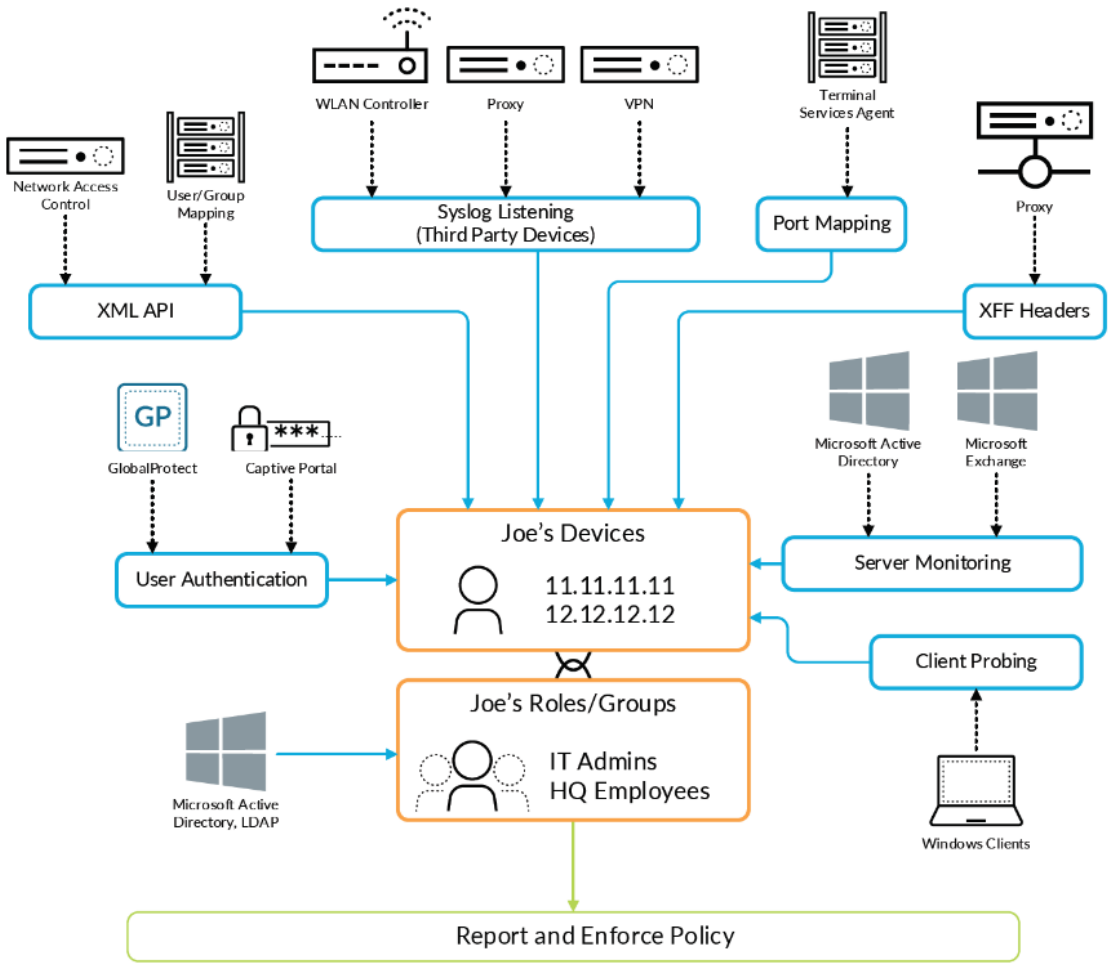
- **Policy control.** Binding user information to the security policy helps organizations to safely enable applications or specific application functions, while reducing the administrative effort associated with employee moves, adds, and changes.
- **Logging and reporting.** If a security incident occurs, forensics analysis and reporting can include user information, which provides a more complete picture of the incident.

User-ID in action

User-ID seamlessly integrates Palo Alto Networks next-generation firewalls with a wide range of user repositories and terminal services environments. Depending on the network environment, multiple techniques can be configured to accurately map the user identity to an IP address. Events include authentication events, user authentication, terminal services monitoring, client probing, directory services integration, and a powerful XML API (see Figure 2-9).

Figure 2-9

User-ID integrates enterprise directories for user-based policies, reporting, and forensics.



After the applications and users are identified, full visibility and control within the Application Command Center (ACC), policy editing, and logging and reporting are available. User-ID tools and techniques include:

- **User authentication.** This technique allows organizations to configure a challenge-response authentication sequence to collect user and IP address information, using the following tools:
    - **Authentication Portal.** In cases where administrators need to establish rules under which users are required to authenticate to the firewall before accessing the internet, the Authentication Portal can be deployed. Authentication Portal is used in cases where the user cannot be identified using other mechanisms. In addition to an explicit username and password prompt, Authentication Portal can also be configured to send an NT LAN Manager (NTLM) authentication request to the web browser to make the authentication process transparent to the user.
    - **Prisma Access.** Users logging in to the network with Prisma Access (discussed in Section 3.5.2) provide user and host information to the next-generation firewall, which, in turn, can be used for policy control.
- **Server monitoring.** Monitoring of the authentication events on a network allows User-ID to associate a user with the IP address of the device from which the user logs in to enforce policy on the firewall. User-ID can be configured to monitor authentication events for:
    - **Microsoft Active Directory.** User-ID constantly monitors domain controller event logs to identify users when they log in to the domain. When a user logs in to the Windows domain, a new authentication event is recorded on the corresponding Windows domain controller. By remotely monitoring the authentication events on Windows domain controllers, User-ID can recognize authentication events to identify users on the network for creation and enforcement of policy.
    - **Microsoft Exchange Server.** User-ID can be configured to constantly monitor Microsoft Exchange login events produced by clients accessing their email. Using this technique, even macOS, Apple iOS, and Linux/Unix client systems that do not directly authenticate to Active Directory can be discovered and identified.
    - **Novell eDirectory.** User-ID can query and monitor login information to identify users and group memberships via standard Lightweight Directory Access Protocol (LDAP) queries on eDirectory servers.

- **Client probing and terminal services.** This technique enables organizations to configure User-ID to monitor Windows clients or hosts to collect the identity and map it to the IP address. In environments where the user identity is obfuscated by Citrix XenApp or Microsoft Terminal Services, the User-ID Terminal Services agent can be deployed to determine which applications are being accessed by users. The following techniques are available to enable this:
    - **Client probing.** If a user cannot be identified via monitoring of authentication events, User-ID actively probes Microsoft Windows clients on the network for information about the currently logged-on user. With client probing, laptop users who often switch from wired to wireless networks can be reliably identified.
    - **Host probing.** User-ID can also be configured to probe Windows servers for active network sessions of a user. As soon as a user accesses a network share on the server, User-ID identifies the origin IP address and maps it to the username provided to establish the session.
    - **Terminal services.** Users sharing IP addresses while working on Microsoft Terminal Services or Citrix can be identified. Every user session is assigned a certain port range on the server, which is completely transparent to the user and allows the next-generation firewall to associate network connections with users and groups sharing one host on the network.
- **XML API.** In some cases, organizations may already have a user repository or an application that is used to store information about users and their current IP address. In these scenarios, the XML API within User-ID enables rapid integration of user information with security policies. The XML API provides a programmatic way to map users to IP addresses through integrations with partner technologies, such as Aruba ClearPass and Aruba Mobility Controllers. Use of the XML API to collect user and IP address information includes:
    - **Wireless environments.** Organizations using 802.1*x* to secure corporate wireless networks can leverage a syslog-based integration with the User-ID XML API to identify users as they authenticate to the wireless infrastructure.
    - **Proxies.** Authentication prompted by a proxy server can be provided to User-ID via its XML API by parsing the authentication log file for user and IP address information.

- **Network access control (NAC).** The XML API enables organizations to harvest user information from NAC environments. As an example, Bradford Networks, a NAC solution provider, uses the User-ID XML API to populate user logins and logouts of its 802.1*x* solution. This integration enables organizations to identify users as soon as they connect to the network and set user-based enablement policies.

- **Syslog listener.** In environments with existing network services that authenticate users – for example, wireless controllers, 802.1*x*, or NAC products – User-ID can monitor syslog messages for user mapping. Extensible syslog filters control the parsing of syslog messages. Syslog filters can be user-defined, but several predefined filters are available, including those for Blue Coat proxy, wireless local-area networks (WLANs), and Pulse Policy Secure.

To enable organizations to specify security rules based on user groups and to resolve the group members automatically, User-ID integrates with directory servers by using a standards-based protocol and a flexible configuration. After integration with the directory server is configured, the firewall automatically retrieves user and user group information and keeps the information updated to automatically adjust to changes in the user base or organization.

After User-ID gathers the user information, the next-generation firewall uses LDAP to obtain group information for that user. Also, as in the case of user mapping, the XML API can serve as a programmatic interface for a flexible group mapping ability. With group mapping, User-ID can express security policies in terms of groups, enabling existing policies to update dynamically as User-ID adds or removes users from groups.

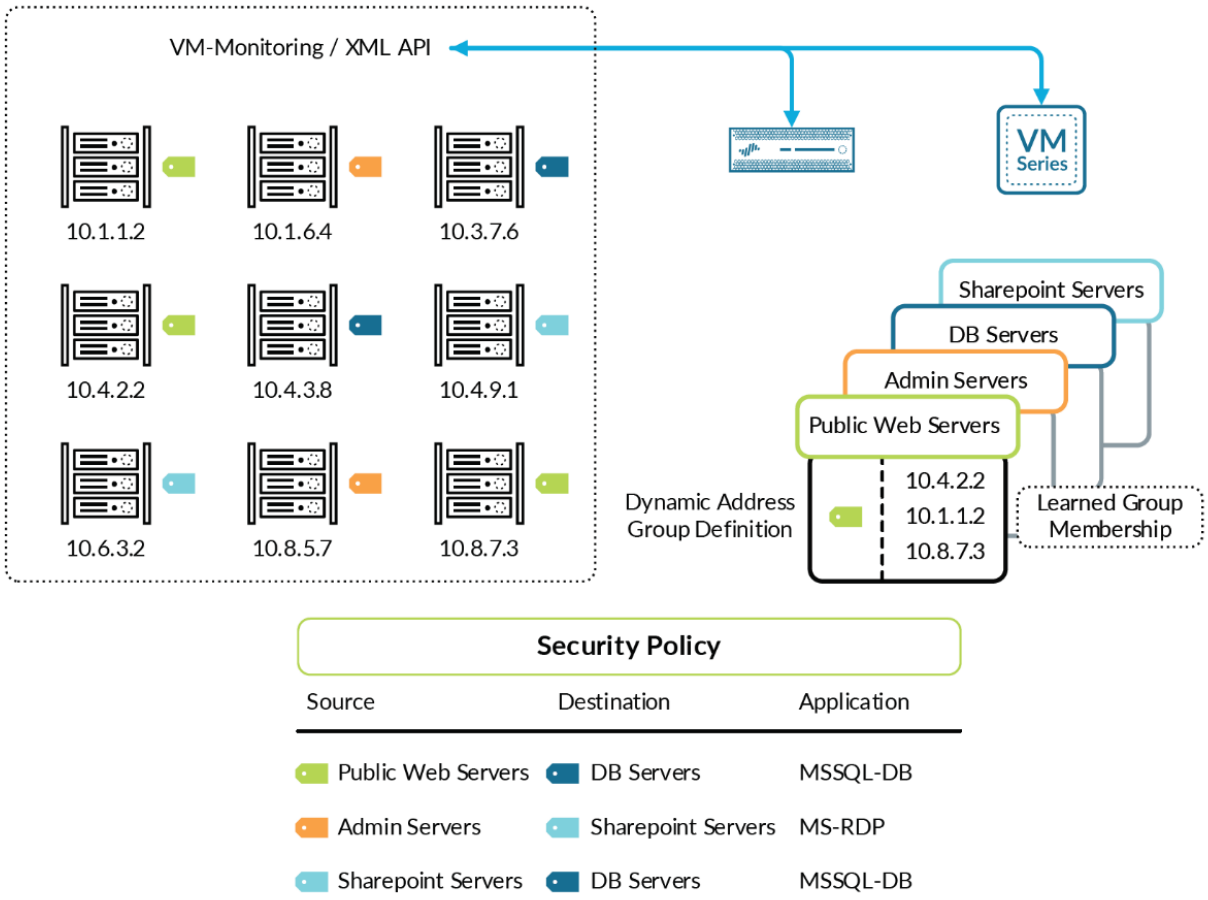
User-ID gives you only half the picture when tying IP addresses to specific users. Servers and many other devices cannot use a user to identify their security access requirements. Dynamic address groups (DAGs) enable policy creation that automatically adapts to server additions, moves, or deletions. DAGs also enable the flexibility to apply security policy to the device based on its role on the network.

A DAG uses tags as a filtering criterion in order to determine its members. You can define tags statically or register them dynamically. You can dynamically register the IP address and associated tags for a device on the firewall by using the XML API or the VM Monitoring agent on the firewall; each registered IP address can have multiple tags. Within 60 seconds of the API call, the firewall registers the IP address and associated tags and automatically updates the membership information for the DAGs.

Because the members of a DAG are automatically updated, you can use address groups to adapt to changes in your environment without relying on a system administrator to make policy changes and committing them (see Figure 2-10).

Figure 2-10

Dynamic address groups (DAGs)



Visibility into a user's activity

The power of User-ID becomes evident when App-ID finds a strange or unfamiliar application on the network. An administrator can use either the ACC or the log viewer to identify the application, who is using it, the bandwidth and session consumption, the sources and destinations of the application traffic, and any associated threats.

Visibility into the application activity at a user level, instead of just at an IP address level, allows organizations to enable the applications traversing the network more effectively. Administrators can align application usage with business unit requirements and, if appropriate,

can choose to inform the user that they are in violation of policy. They can also take the more direct approach of blocking the user's application usage outright.

User-based policy control

User-based policy controls can be created based on the application, category and subcategory, underlying technology, or application characteristics. Policies can be used to safely enable applications based on users or groups in either an outbound or an inbound direction.

User-based policies might include:

- Enable only the IT department to use tools such as SSH, Telnet, and FTP on their standard ports.
- Allow the Help Desk Services group to use Yahoo Messenger.
- Allow Facebook for all users, allow only the Marketing group to use Facebook-posting, and block the use of Facebook applications for all users.

Policy Optimizer

Policy Optimizer can help organizations migrate from legacy firewall rule configurations to application-based rules through App-ID. This capability strengthens the security posture by using App-ID to close any security gaps and minimizes configuration errors – a leading cause of breaches. Policy Optimizer analyzes application use and recommends policy rules that reduce exposure and risk.

Policy Optimizer identifies port-based rules so that they can be converted to application-based rules. Converting from port-based to application-based rules improves the overall security posture because you can whitelist the applications you want to permit and then deny all others. Policy Optimizer makes it simple to prioritize which of the port-based rules to migrate first, identify application-based rules that allow applications you don't use, and analyze each of the rules' usage characteristics, such as hit count.

2.6.1.3 Content identification

Content identification infuses next-generation firewalls with capabilities not possible in legacy port-based firewalls. Application identification eliminates threat vectors through the tight control of all types of applications. This capability immediately reduces the attack surface of the network, after which all allowed traffic is analyzed for exploits, malware, dangerous URLs, and dangerous or restricted files or content. Content identification then goes beyond stopping known threats to proactively identifying and controlling unknown malware, which is often used as the leading edge of sophisticated network attacks.

Threat prevention

Enterprise networks are facing a rapidly evolving threat landscape full of modern applications, exploits, malware, and attack strategies that can avoid traditional methods of detection. Threats are delivered via applications that dynamically hop ports, use non-standard ports, tunnel within other applications, or hide within proxies, SSL, or other types of encryption. These techniques can prevent traditional security solutions such as IPS and firewalls from ever inspecting the traffic, thus enabling threats to flow across the network easily and repeatedly. Also, enterprises are exposed to targeted and customized malware, which may pass undetected through traditional anti-malware solutions.

Palo Alto Networks Content-ID addresses these challenges with unique threat prevention capabilities not found in traditional security solutions. First, the next-generation firewall removes the methods that threats use to hide from security through the complete analysis of all traffic on all ports, regardless of any evasion, tunneling, or circumvention techniques used. No threat prevention solution will be effective if it does not have visibility into the traffic. Palo Alto Networks ensures that visibility through the identification and control of all traffic, using the following tools and techniques:

- **Application decoders.** Content-ID leverages the more than 100 application and protocol decoders in App-ID to look for threats hidden within application data streams. This tool enables the firewall to detect and prevent threats tunneled within approved applications that would bypass traditional IPS or proxy solutions.

- **Uniform threat signature format.** Rather than use a separate set of scanning engines and signatures for each type of threat, Content-ID leverages a uniform threat engine and signature format to detect and block a wide range of malware C2 activity and vulnerability exploits in a single pass.

- **Vulnerability attack protection (IPS).** Robust routines for traffic normalization and defragmentation are joined by protocol-anomaly, behavior-anomaly, and heuristic detection mechanisms to provide protection from the widest range of both known and unknown threats.

- **Cloud-based intelligence.** For unknown content, WildFire (discussed in Section 2.6.2.4) provides rapid analysis and a verdict that the firewall can leverage.

- **SSL decryption.** More and more web traffic connections are encrypted with SSL by default, which can provide some protection to end users, but SSL also can provide attackers with an encrypted channel to deliver exploits and malware. Palo Alto

Networks ensures visibility by giving security organizations the flexibility to, by policy, granularly look inside SSL traffic based on application or URL category.

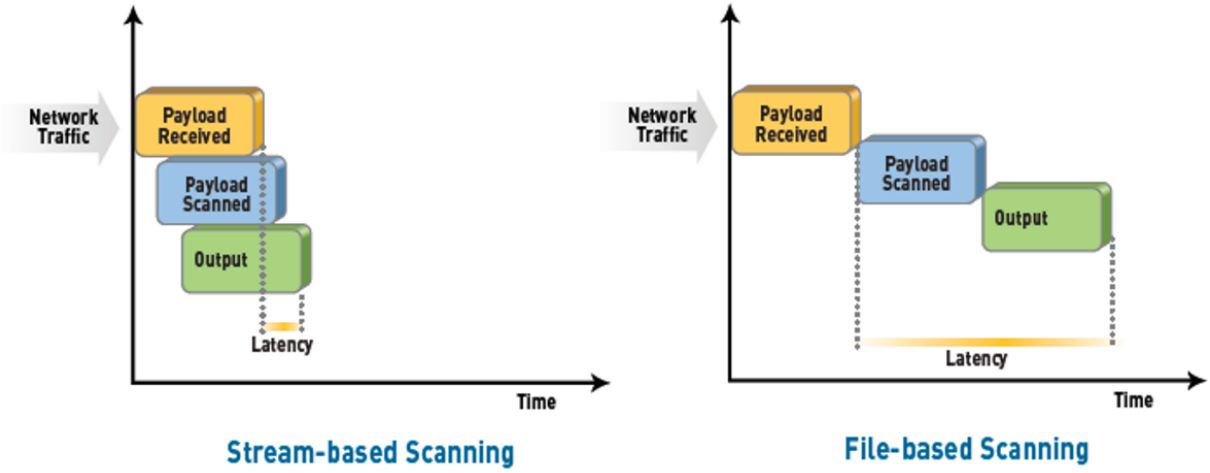
- **Control of circumventing technologies.** Attackers and malware have increasingly turned to proxies, anonymizers, and a variety of encrypted proxies to hide from traditional network security products. Palo Alto Networks provides the ability to tightly control these technologies and limit them to approved users, while blocking unapproved communications that could be used by attackers.

Stream-based malware scanning

Prevention of known malware is performed through the use of stream-based scanning, a technique that begins scanning as soon as the first packets of the file are received, as opposed to waiting until the entire file is loaded into memory to begin scanning. Stream-based scanning minimizes performance and latency issues by receiving, scanning, and sending traffic to its intended destination immediately without having to first buffer and then scan the file (see Figure 2-11).

Figure 2-11

Stream-based scanning helps minimize latency and maximize throughput performance.



Intrusion prevention

Content-ID protects networks from all types of vulnerability exploits, buffer overflows, DoS attacks, and port scans that lead to the compromise of confidential and sensitive enterprise information. IPS mechanisms in Content-ID include:

- Protocol decoders and anomaly detection
- Stateful pattern matching

- Statistical anomaly detection
- Heuristic-based analysis
- Invalid or malformed packet detection
- IP defragmentation and TCP reassembly
- Custom vulnerability and spyware phone-home signatures

Traffic is normalized to eliminate invalid and malformed packets, while TCP reassembly and IP defragmentation are performed to ensure the utmost accuracy and protection despite any packet-level evasion techniques.

File and data filtering

File and data filtering takes advantage of in-depth application inspection and enables enforcement of policies that reduce the risk of unauthorized information transfer or malware propagation. File and data filtering capabilities in Content-ID include:

- **File blocking by type.** Control the flow of a wide range of file types by looking deep within the payload to identify the file type (as opposed to looking only at the file extension).
- **Data filtering.** Control the transfer of sensitive data patterns such as credit card numbers and Social Security numbers in application content or attachments.
- **File transfer function control.** Control the file transfer functionality within an individual application, which allows application use while preventing undesired inbound or outbound file transfer.

2.6.1.4 Log correlation and reporting

Powerful log filtering enables administrators to quickly investigate security incidents by correlating threats with applications and user identity. The ACC provides a comprehensive view of current and historical data – including network activity, application usage, users, and threats – in a highly visual, fully customizable, and easy-to-use interactive format. This visibility enables administrators to make informed policy decisions and respond quickly to potential security threats.

The ACC provides a tabbed view of network activity, threat activity, and blocked activity, and each tab includes pertinent widgets for better visualization of traffic patterns on the network (see Figure 2-12).

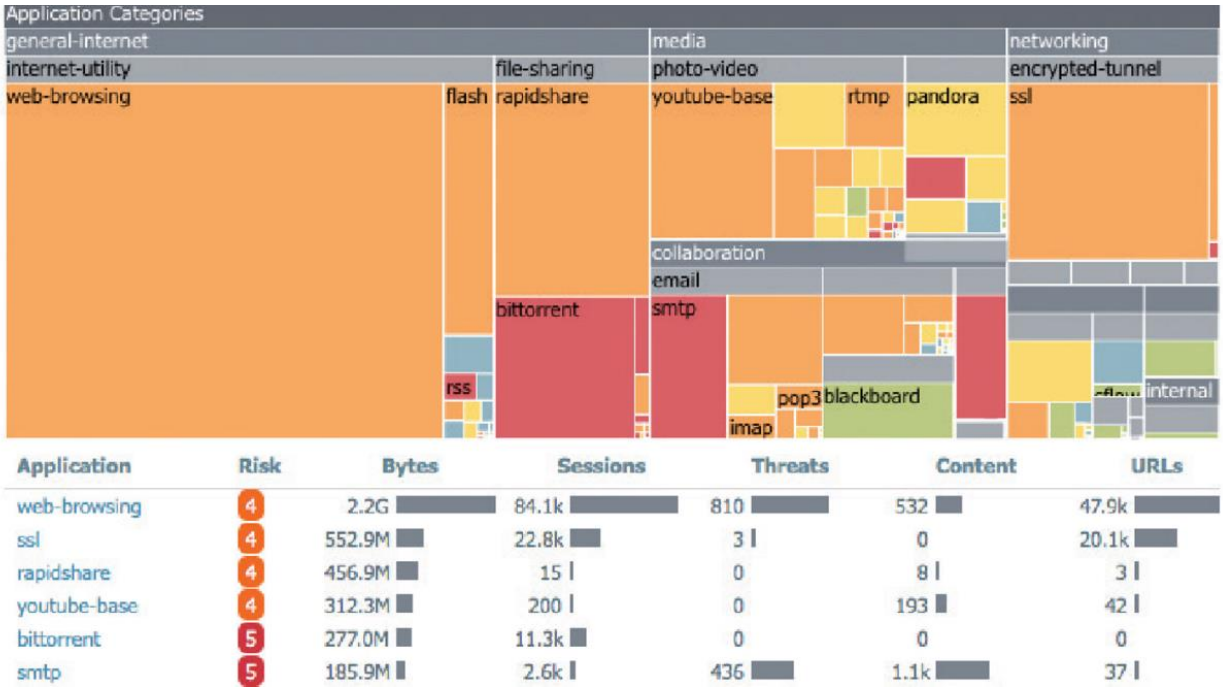
Figure 2-12

The ACC provides a highly visual, interactive, and customizable security management dashboard.

Figure 2-13 shows a core widget of the ACC, the Application Usage widget. In this case, the widget shows application traffic in bytes. Applications (colored boxes) are grouped in application categories (gray bars). The size of each box indicates how much traffic a given application consumed during the selected time frame. The color of the box indicates the risk level of an application, with red being critical, orange medium, and blue the lowest risk. The tabular listing below the graph shows additional information, such as the number of sessions, threats detected, content or files included, and URLs accessed by these applications.

Figure 2-13

The ACC Application Usage widget displays application traffic by type, amount, risk, and category.



Application	Risk	Bytes	Sessions	Threats	Content	URLs
web-browsing	4	2.2G	84.1k	810	532	47.9k
ssl	4	552.9M	22.8k	3	0	20.1k
rapidshare	4	456.9M	15	0	8	3
youtube-base	4	312.3M	200	0	193	42
bittorrent	5	277.0M	11.3k	0	0	0
smtp	5	185.9M	2.6k	436	1.1k	37

In Figure 2-14, an ACC widget shows source and destination by region, with a visual display of where traffic is originating and going. The world maps are interactive and provide the ability to get more detail and information about traffic to or from individual countries.

Figure 2-14

Geolocation awareness in the ACC provides valuable information about the source and destination of all application traffic.

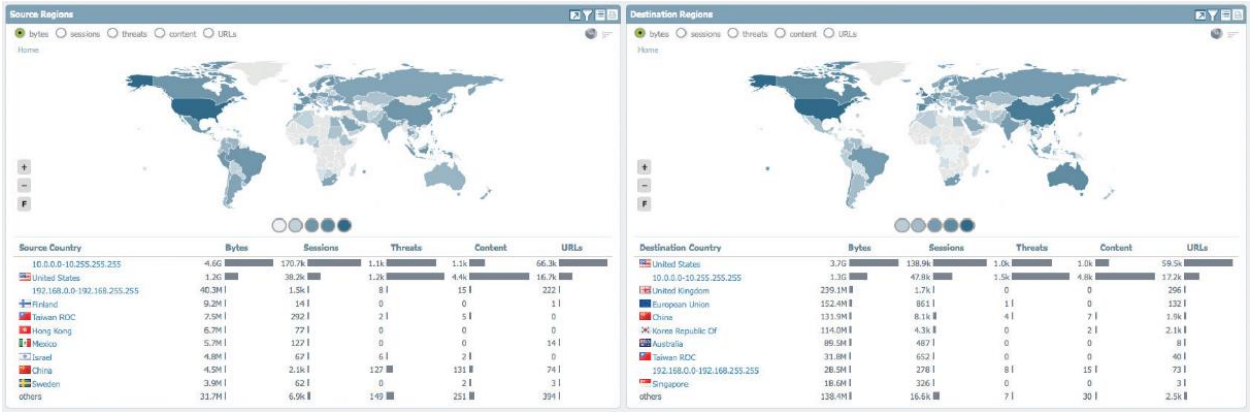
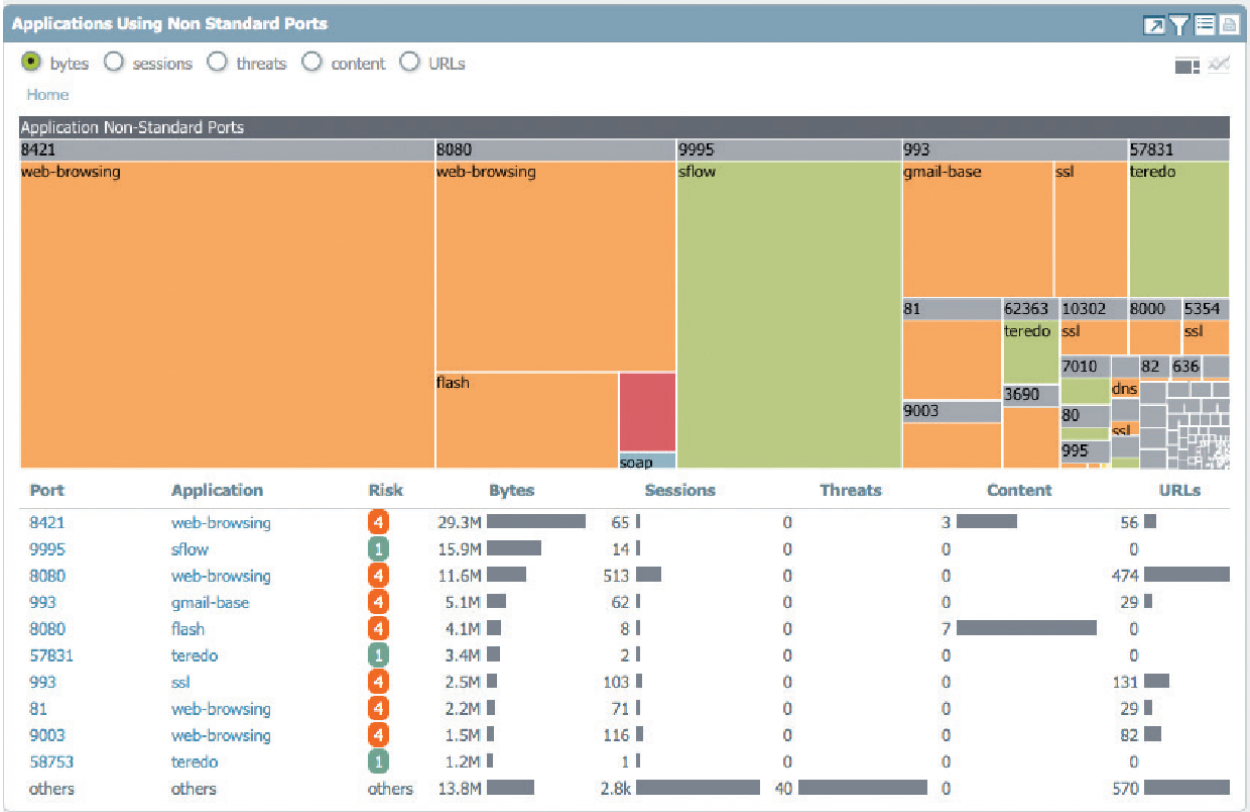


Figure 2-15 shows an ACC widget that demonstrates the power of application control in a next-generation firewall versus a traditional port-based firewall. This widget shows applications with port hopping capabilities using non-standard ports.

Figure 2-15

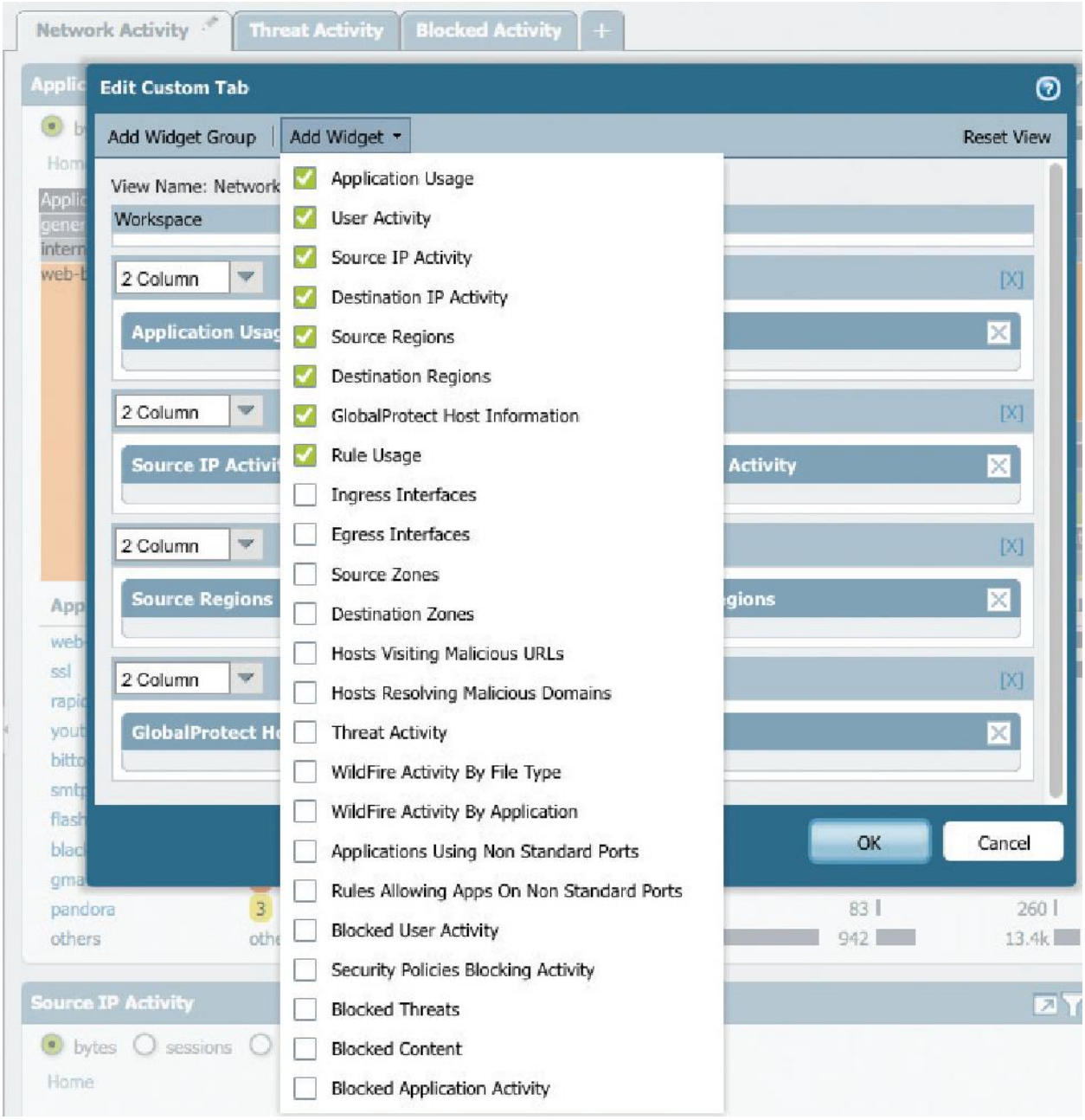
The ACC Applications Using Non Standard Ports widget highlights port hopping and showcases the importance of application versus port control.



Custom tabs can also be created to include widgets that enable administrators to view more specific information. With the ACC, every administrator can customize their own views by selecting predesigned widgets from a drop-down list and building their own user interface (see Figure 2-16).

Figure 2-16

A wide variety of widgets can be selected to customize tabs in the ACC.

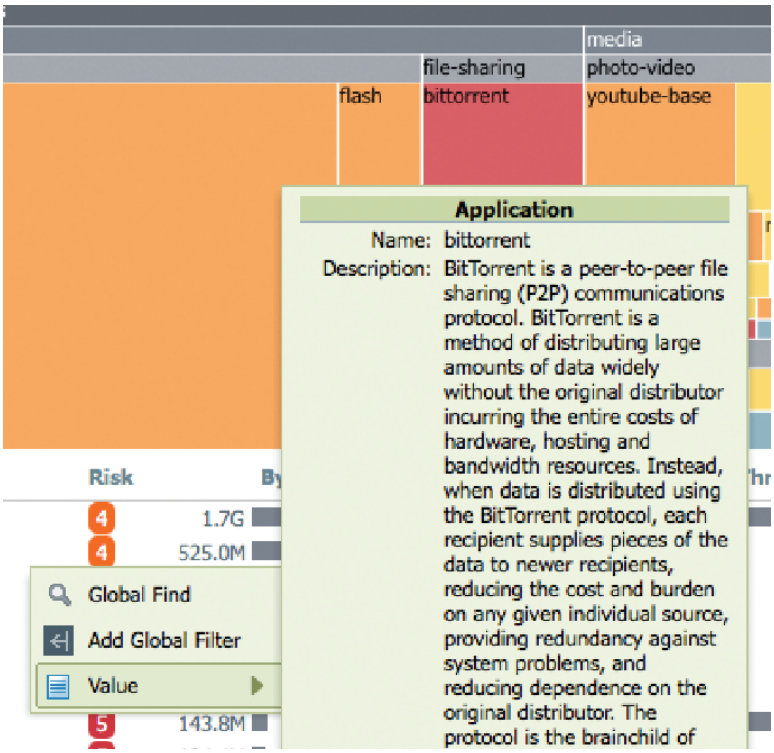


In addition to customizing existing tabs (**Network Activity**, **Threat Activity**, and **Blocked Activity**), administrators can create custom tabs to monitor certain employees, situations, or applications.

With the interactive capabilities of the ACC, you can learn more about applications, URL categories, risk levels, or threats to get a complete picture of network and threat activity (see Figure 2-17).

Figure 2-17

One-click, interactive capabilities provide additional information and the ability to apply any item as a global filter.

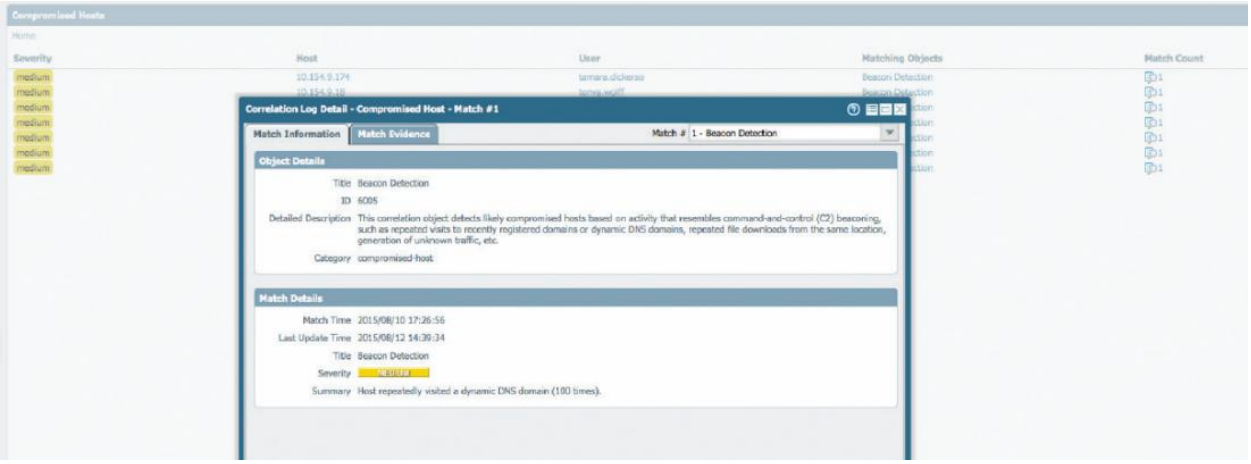


The automated correlation engine in the ACC is an analytics tool that surfaces critical threats that may be hidden in the network. The correlation engine reduces manual data mining and enables faster response times. It scrutinizes isolated events automatically across multiple logs, queries the data for specific patterns, and correlates network events to identify compromised hosts. And it includes correlation objects that are defined by the Palo Alto Networks Malware Research team. These objects identify suspicious traffic patterns, compromised hosts, and other events that indicate a malicious outcome. Some correlation objects can identify dynamic patterns that have been observed from malware samples in WildFire.

Correlation objects trigger correlation events when they match on traffic patterns and network artifacts, indicating a compromised host on your network. In the ACC, correlation triggers are clearly identified and highlighted to enable a fast response (see Figure 2-18).

Figure 2-18

The automated correlation engine automatically highlights compromised hosts in the ACC by correlating indicators of compromise (IoCs).



A log is an automatically generated, timestamped file that provides an audit trail for system events on the firewall or network traffic events that the firewall monitors. Log entries contain *artifacts*, which are properties, activities, or behaviors associated with the logged event, such as the application type or the IP address of an attacker. Each log type records information for a separate event type. For example, the firewall generates a threat log to record traffic that matches a spyware, vulnerability, or virus signature or a DoS attack that matches the thresholds configured for a port scan or host sweep activity on the firewall.

The following logs can be viewed from the **Monitor** tab on Palo Alto Networks next-generation firewalls:

- **Traffic logs.** These logs display an entry for the start and end of each session. Each entry includes the following information: date and time; source and destination zones, addresses, and ports; application name; security rule applied to the traffic flow; rule action (“allow,” “deny,” or “drop”); ingress and egress interface; number of bytes; and session end reason.

- **Threat logs.** These logs display entries when traffic matches one of the Security Profiles attached to a security rule on the firewall. Each entry includes the following information: date and time; type of threat (such as virus or spyware); threat description or URL (**Name** column); source and destination zones, addresses, and ports; application name; alarm action (such as “allow” or “block”); and severity level.

- **URL Filtering logs.** These logs display entries for traffic that matches URL Filtering Profiles attached to security rules. For example, the firewall generates a log if a rule

blocks access to specific websites and website categories or if you configured a rule to generate an alert when a user accesses a specific website.

- **WildFire Submissions logs.** The firewall forwards samples (files and emails links) to the WildFire cloud for analysis based on WildFire Analysis Profiles settings. The firewall generates WildFire Submissions log entries for each sample it forwards after WildFire completes static and dynamic analysis of the sample. WildFire Submissions log entries include the WildFire verdict for the submitted sample.

- **Data Filtering logs.** These logs display entries for the security rules that help prevent sensitive information such as credit card numbers from leaving the area that the firewall protects.

- **Correlation logs.** The firewall logs a correlated event when the patterns and thresholds defined in a correlation object match the traffic patterns on your network.

- **Config logs.** These logs display entries for changes to the firewall configuration. Each entry includes the date and time, the administrator username, the IP address from where the administrator made the change, the type of client (web, CLI, or Panorama), the type of command executed, the command status (succeeded or failed), the configuration path, and the values before and after the change.

- **System logs.** These logs display entries for each system event on the firewall. Each entry includes the date and time, event severity, and event description.

- **HIP Match logs.** The Prisma Access Host Information Profile (HIP) feature enables you to collect information about the security status of the end devices accessing your network (such as whether they have disk encryption enabled). The firewall can allow or deny access to a specific host based on adherence to the HIP-based security rules you define. HIP Match logs display traffic flows that match a HIP Object or HIP Profile that you configured for the rules.

- **Alarms logs.** An alarm is a firewall-generated message that indicates that the number of events of a particular type (for example, encryption and decryption failures) has exceeded the threshold configured for that event type.

- **Unified logs.** Unified logs are entries from the Traffic, Threat, URL Filtering, WildFire Submissions, and Data Filtering logs displayed in a single view. The Unified log view enables you to investigate and filter the latest entries from different log types in one place, instead of searching through each log type separately.

The reporting capabilities on the Palo Alto Networks next-generation firewall enable you to monitor your network health, validate your policies, and focus your efforts on maintaining network security. The following report types are available:

- **Predefined reports** allow you to view a summary of the traffic on your network. Predefined reports are available in four categories: Applications, Traffic, Threat, and URL Filtering.
- **User or group activity reports** allow you to schedule or create an on-demand report on the application use and URL activity for a specific user or for a user group. The report includes the URL categories and an estimated browse-time calculation for individual users.
- **Custom reports** can be created and scheduled to show exactly the information you want to see by filtering on conditions and columns to include. You can also include query builders for more specific details in report data.
- **PDF summary reports** aggregate up to 18 predefined or custom reports and graphs from Threat, Application, Trend, Traffic, and URL Filtering categories into one PDF document.
- **Botnet reports** allow you to use behavior-based mechanisms to identify potential botnet-infected hosts in the network.
- **Report groups** combine custom and predefined reports into report groups and compile a single PDF document that is emailed to one or more recipients.

Reports can be generated on demand or on a recurring schedule, and they can be scheduled for email delivery.

2.6.1.5 Implementing Zero Trust with next-generation firewalls

Companies are often reluctant to begin the Zero Trust journey because they believe it is difficult, costly, and disruptive. Twentieth-century design paradigms can create problems when designing a twenty-first-century Zero Trust network. However, building Zero Trust networks is actually much simpler than building legacy twentieth-century hierarchical networks. Because most of us learned to design networks from the outside in, based on classifying users as “trusted” and “untrusted” – an approach that has since proven unsecure – we struggle to adapt our design thinking to the Zero Trust methodology.

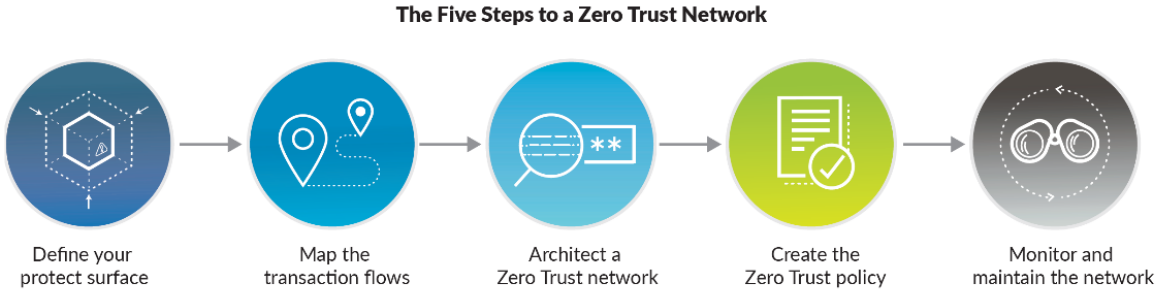
It’s not necessary to rip and replace your existing network to deploy a Zero Trust network. Zero Trust augments your existing network, with each Zero Trust network designed for a specific

protect surface. The Zero Trust network is interconnected with your existing network to take advantage of the technology you already have. Then, over time, you iteratively move your additional datasets, applications, assets, or services from your legacy network to your Zero Trust network. This phased approach helps make deploying Zero Trust networks manageable, cost-effective, and non-disruptive.

The following five-step methodology describes a Zero Trust deployment with next-generation firewalls and other tightly integrated Palo Alto Networks security solutions (see Figure 2-19).

Figure 2-19

The Palo Alto Networks Zero Trust methodology



Step 1: Define your protect surface

When defining the protect surface, you need to consider all critical data, application, assets, or services (DAAS). Your protect surface could include:

- **Data.** Payment card information (PCI), protected health information (PHI), personally identifiable information (PII), and intellectual property
- **Applications.** Off-the-shelf or custom software
- **Assets.** Supervisory control and data acquisition (SCADA) controls, point-of-sale terminals, medical equipment, manufacturing assets, and IoT devices
- **Services.** Domain Name System (DNS), Dynamic Host Configuration Protocol (DHCP), and Active Directory

Palo Alto Networks next-generation firewalls, in physical or virtualized form, provide comprehensive Layer 7 visibility to help you determine your data, applications, assets, and service profile. Palo Alto Networks also has extensive partnerships with leading third-party companies to help with additional data and asset discovery. Cortex XDR (discussed in Section 4.4.1) detection and response utilizes network, cloud, and endpoints as sensors, feeding data into Cortex Data Lake (discussed in Section 4.4.4) to provide visibility into the activity of users,

devices, applications, and services for greater insight into the individual protect surfaces across your enterprise environment.

Step 2: Map the transaction flows

To properly design a network, it's critical to understand how systems should work. The way traffic moves across the network (specific to the data in the protect surface) determines how it should be protected. This understanding comes from scanning and mapping the transaction flows inside your network in order to determine how various data, application, asset, and service components interact with other resources on your network.

It's common to approximate flows by documenting what you know about how specific resources interact. Even without a complete picture, this information still provides valuable data so that you don't arbitrarily implement controls with zero insight.

Zero Trust is a flow-based architecture. After you understand how your systems are designed to work, the flow maps will tell you where you need to insert controls.

Remember that Zero Trust is an iterative process. Start with what you know. As you move through the steps in this methodology, you'll gather more information that will enable more granularity in your design. You shouldn't delay your Zero Trust initiative just because you don't have perfect information.

Palo Alto Networks next-generation firewalls deliver deep, application-layer visibility with granular insight into traffic flows. Policy Optimizer (discussed in Section 2.6.1.2) gives deep visibility into applications to help you prioritize rule migration, identify rules that allow unused or overprovisioned applications, and analyze rule usage characteristics.

Additionally, Cortex Data Lake collects telemetry from the network via next-generation firewall appliances, the cloud via VM-Series virtualized next-generation firewalls (discussed in Section 2.6.1.6), and endpoints via Cortex XDR. With this data centralized, Cortex XDR taps into Cortex Data Lake to validate established interaction and provide details around that interaction to help refine the use of communication and understanding of the flow.

Step 3: Architect a Zero Trust network

Traditionally, the first step of any network design is to architect it. Individuals get "reference architectures" for the network and must work to make them usable for their business. In the Zero Trust journey, architecting the network is the third step. Further, Zero Trust networks are bespoke, not some universal design. After the protect surface is defined and the flows mapped, the Zero Trust architecture will become apparent.

The architectural elements begin with deploying a next-generation firewall as a segmentation gateway to enforce granular Layer 7 access as a micro-perimeter around the protect surface. With this architecture, each packet that accesses a resource inside the protect surface will pass through a next-generation firewall so that Layer 7 policy can be enforced, simultaneously controlling and inspecting access. There is a significant misunderstanding that Zero Trust is only about access control: Least-privileged access control is only one facet of Zero Trust. Another facet is the inspection and logging of every single packet, all the way through Layer 7, to determine if packets are clean. This determination is made by inspecting all network traffic for malicious content with multiple integrated security services, including IPS, sandboxing, DNS security, URL filtering, and data loss prevention (DLP) capabilities.

Palo Alto Networks next-generation firewalls take advantage of App-ID, User-ID, and Content-ID to define authoritative Layer 7 policy controls and prevent compromise of protect surfaces. Because these segmentation gateways are offered in both physical and virtual form factors, this architectural model can work everywhere you may have a protect surface, whether in on-premises or off-premises physical data centers, or in private, public, or hybrid cloud environments.

Endpoint security, such as Cortex XDR (discussed in Section 4.4.1), can prevent compromise of the protect surface by known and unknown threats, whether from malware, fileless attacks, or exploits. Secure access offerings, such as Prisma Access (discussed in Section 3.5.2), extend the policy of each micro-perimeter down to the endpoints attempting to access protect surface resources.

The Security Operating Platform delivers telemetry from all core Palo Alto Networks technologies to Cortex Data Lake (discussed in Section 4.4.4), enabling machine learning based policy optimization and automation via Cortex XDR for improvement in later stages of your deployment.

The architecture would still be incomplete without important third-party offerings. Palo Alto Networks integrates with multiple multi-factor authentication (MFA) providers to add fidelity to User-ID. To round out and simplify Zero Trust architectures, a powerful API provides deep integrations with more than 250 third-party partners, including anti-spam/anti-phishing technologies, DLP systems, software-defined wide-area networks (SD-WAN), and wireless offerings.

Step 4: Create the Zero Trust policy

After you've architected your Zero Trust network, you need to create the supporting Zero Trust policies, following the Kipling Method, to answer the who, what, when, where, why, and how of your network and policies. For one resource to talk to another, a specific rule must whitelist

that traffic. The Kipling Method of creating policy enables Layer 7 policy for granular enforcement so that only known allowed traffic or legitimate application communication is allowed in your network. This process significantly reduces the attack surface while also reducing the number of port-based firewall rules enforced by traditional network firewalls. With the Kipling Method, you can easily write policies by answering:

- **Who** should be accessing a resource? This defines the "asserted identity."
- **What** application is the asserted identity of the packet using to access a resource inside the protect surface?
- **When** is the asserted identity trying to access the resource?
- **Where** is the packet destination? A packet's destination is often automatically pulled from other systems that manage assets in an environment, such as from a load-balanced server via a virtual IP address.
- **Why** is this packet trying to access this resource within the protect surface? This question relates to data classification, where metadata automatically ingested from data classification tools helps make your policy more granular.
- **How** is the asserted identity of a packet accessing the protect surface via a specific application?

To simplify the process, you should create policies primarily on your segmentation gateways' centralized management tool. Panorama (discussed in Section 2.6.3) provides this functionality, and Panorama is where the Kipling Method is applied.

Palo Alto Networks next-generation firewall technology and unique features enable you to write policies that are easy to understand and maintain while providing maximum security transparency to your end users. User-ID helps define the who, App-ID helps define the what, and Content-ID helps define the how, all of which is enforced throughout your deployment, including by the WildFire malware prevention service, as well as by the Threat Prevention, URL Filtering, and DNS Security services. PAN-OS delivers enhanced policy creation capability, notably through Policy Optimizer, which continuously helps you understand how to increase the fidelity of your Zero Trust policy. Additionally, you can create policies for Prisma SaaS based on how SaaS applications are accessed.

Step 5: Monitor and maintain the network

The last step in this iterative process is to monitor and maintain your network, which means continuously looking at all internal and external logs through Layer 7 and focusing on the

operational aspects of Zero Trust. Inspecting and logging all traffic on your network is a pivotal facet of Zero Trust.

It's important to send the system as much telemetry as possible about your environment. This data will give you new insights into how to improve your Zero Trust network over time. The more your network is attacked, the stronger it will become, with greater insight into making policies more secure. Additional data provides insight into the protect surface – such as what you should include in it and the interdependencies of data within it – that can inform architectural tweaks to further enhance your security.

All telemetry generated by Palo Alto Networks endpoint, network, and cloud security technologies is sent to Cortex Data Lake, where the data is stitched together to enable machine learning based policy optimization and analytics.

Next-generation firewall and VM-Series data is consolidated into a singular view under Panorama, which raises an alert when a malicious or suspicious occurrence should be investigated.

Cortex XSOAR Threat Intelligence Management (TIM) (discussed in Section 4.4.3) takes a unique approach to native threat intelligence management, unifying aggregation, scoring, and sharing with playbook-driven automation.

Prisma Cloud (discussed in Section 3.4) provides public cloud security and compliance monitoring, scanning all audit and flow logs across multicloud environments for root user and overly permissive administrator activities. Prisma Cloud builds a deep contextual understanding of your cloud environment, allowing detection of user anomalies – based on activity and location – that could signal compromised credentials, brute-force attacks, and other suspicious activities. Prisma Cloud also correlates threat intelligence data to provide visibility into suspicious IP addresses and host vulnerabilities across your resources, which can quickly be isolated to avoid additional exposure. This data provides insight that enables you to fine-tune Zero Trust privileges.

Cortex XDR (discussed in Section 4.4.1) takes advantage of Cortex Data Lake (discussed in Section 4.4.4) to create profiles of users and devices, acting as a baseline of normal use. This baseline allows the behavioral analytics engine to detect threats based on anomalies targeting your protect surface. In evaluating current or additional protect surface policies, Cortex XDR allows you to search the telemetry within Cortex Data Lake for communication and interactions between entities. You can also analyze the telemetry to prove the condition or get valuable insight into how your policy should be modified. In rare instances, the search can identify an unknown threat vector not factored into the protect surface. Cortex XDR will then facilitate a

deep investigation of the newfound threat so that you can uncover what occurred and react accordingly.

2.6.1.6 Next-generation firewall deployment options

The Palo Alto Networks family of next-generation firewalls includes physical appliances, virtualized firewalls, and 5G-ready firewalls.

Physical appliances (PA-Series)

The full range of Palo Alto Networks physical next-generation firewalls is easy to deploy into your organization's network. They are purposefully designed for simplicity, automation, and integration. PA-Series firewalls support a variety of data center and remote branch deployment use cases. Available PA Series firewalls include (see Figure 2-20):

- **PA-7000 Series.** The PA-7000 Series next-generation firewalls enable enterprise-scale organizations and service providers to deploy security in high-performance environments, such as large data centers and high-bandwidth network perimeters. Designed to handle growing throughput needs for application-, user-, and device-generated data, these systems offer amazing performance, prevention capabilities to stop the most advanced cyberattacks, and high-throughput decryption to stop threats hiding under the veil of encryption. Built to maximize security-processing resource utilization and automatically scale as new computing power becomes available, the PA-7000 Series offers simplicity defined by a single-system approach to management and licensing.

- **PA-5200 Series.** The PA-5200 Series next-generation firewalls – comprising the PA-5280, PA-5260, PA-5250, and PA-5220 firewalls – are ideal for high-speed data center, internet gateway, and service provider deployments. The PA-5200 Series delivers up to 64Gbps of throughput, using dedicated processing and memory, for the key functional areas of networking, security, threat prevention, and management.

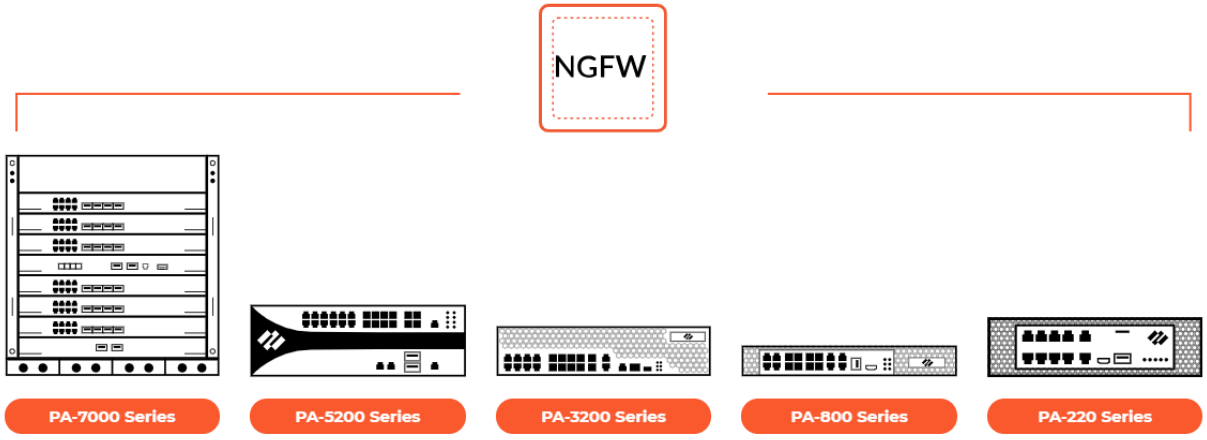
- **PA-3200 Series.** The PA-3200 Series next-generation firewalls – comprising the PA-3260, PA-3250, and PA-3220 – are targeted at high-speed internet gateway deployments. PA-3200 Series appliances secure all traffic (including encrypted traffic), using dedicated processing and memory for networking, security, threat prevention, and management.

- **PA-800 Series.** The PA-800 Series next-generation firewalls – comprising the PA-850 and PA-820 firewalls – are designed to provide secure connectivity for organizations' branch offices as well as midsize businesses.

- **PA-220.** The PA-220 firewall brings next-generation firewall capabilities to distributed enterprise branch offices, retail locations, and midsize businesses in a small form factor.
- **PA-220R.** The PA-220R firewall is a ruggedized next-generation firewall that secures industrial and defense networks in a range of harsh environments, such as utility substations, power plants, manufacturing plants, oil and gas facilities, building management systems, and healthcare networks.

Figure 2-20

Strata Next-Generation Firewalls



Virtualized firewalls (VM-Series)

VM-Series virtual firewalls provide all the capabilities of Palo Alto Networks next-generation physical hardware firewalls (PA-Series) in a virtual machine form factor. VM-Series form factors support a variety of deployment use cases, including:

- **Micro-segmentation.** VM-Series virtual firewalls reduce your environment's attack surface by enabling granular segmentation and micro-segmentation. Threat prevention capabilities ensure that when threats do enter the environment, they are quickly identified and stopped before they can exfiltrate data, deliver malware or ransomware payloads, or cause other damage.
- **Multicloud and hybrid cloud.** VM-Series virtual firewalls eliminate the need for multiple security tool sets by providing comprehensive visibility and control across multicloud and hybrid cloud environments – including Amazon Web Services (AWS), Google Cloud Platform (GCP), Microsoft Azure, and Oracle Cloud – and just as effortlessly in software-defined networks and virtualized environments, all managed from a single console.

- **DevOps and CI/CD pipelines.** VM-Series virtual firewalls provide on-demand, elastic scalability to ensure security when and where you need it most. With automated network security, security provisioning can now be integrated directly into DevOps workflows and CI/CD pipelines without slowing the pace of business.

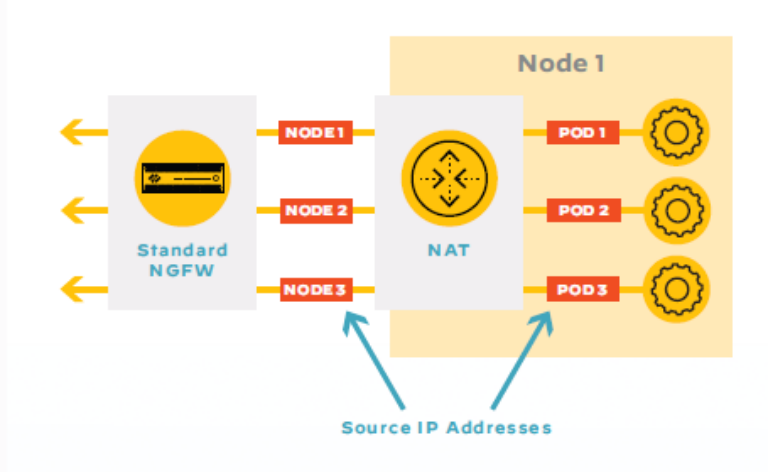
Cloud-native firewalls (CN-Series)

Standard next-generation firewalls play an indispensable role in securing on-premises deployments — few data centers can do without them. However, cloud-native environments pose unique challenges that next-generation firewalls were not designed to handle, especially when it comes to looking inside a Kubernetes environment.

In Kubernetes, pods (collections of containers) run on nodes, either physical or virtual machines. Developers rarely have to deal with nodes explicitly, but nodes impact how firewalls operate. Next-generation firewalls cannot determine which pod is the source of outbound traffic because all source IP addresses are translated to the node IP address. To a traditional firewall, all outbound traffic from the node looks the same (see Figure 2-21).

Figure 2-21

Due to the use of network address translation (NAT) in Kubernetes, all outbound traffic carries the node source IP address.



While Kubernetes creates challenges for traditional security tools, it also presents opportunities to enhance security by taking advantage of native constructs—most notably, namespaces.

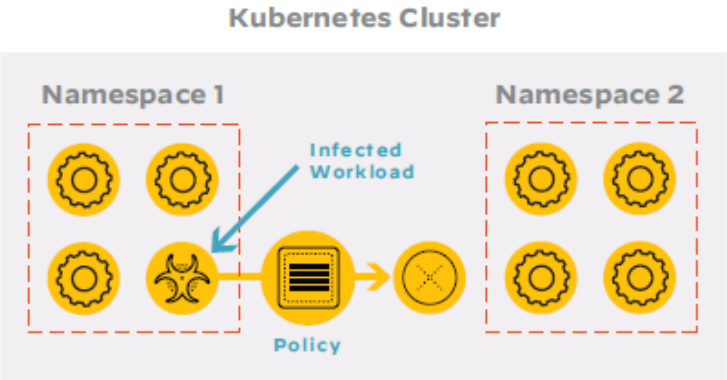
Kubernetes namespaces help to simplify cluster management by making it easier to apply certain policies to some parts of the cluster without affecting others. However, they are also a valuable security tool. Security teams use namespaces to isolate workloads, which reduces the

risk of attacks spreading within a cluster, and establish resource quotas to mitigate the damage that can be caused by a successful cluster breach.[38]

A secure cloud-native architecture requires the ability to secure traffic that crosses namespace boundaries or travels outbound to legacy workloads such as bare metal servers. However, doing so requires knowing the internal state of objects such as namespaces, pods, and containers. Because that information is not available outside the environment, the only effective solution is to take the security solution inside the Kubernetes walls (see Figure 2-22).

Figure 2-22

Security policies based on namespaces prevent spread of exploits within a physical cluster.

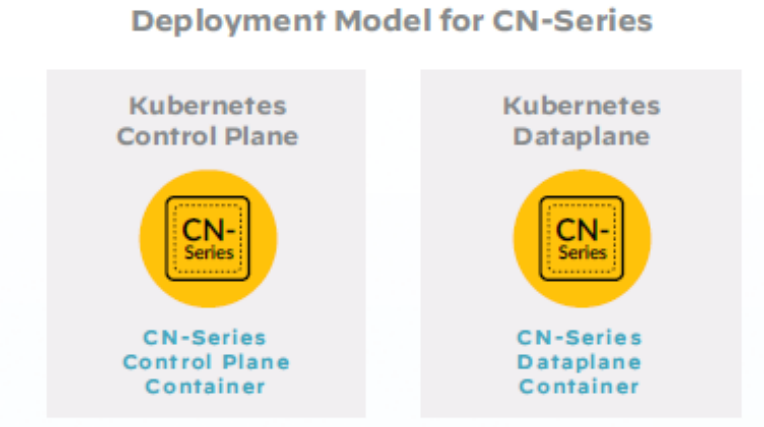


Palo Alto Networks CN-Series next-generation firewalls deploy as two sets of pods: one for the management plane (CN-MGMT), and another for the firewall dataplane (CN-NGFW), as shown in Figure 2-23. The management pod always runs as a Kubernetes service. The dataplane pods can be deployed in two modes: distributed or clustered. In distributed mode, the firewall dataplane runs as a daemon set on each node. Administrators can deploy next-generation firewalls on all cluster nodes with a single command, placing security controls as close to the workloads as possible. In clustered deployment mode, the firewall dataplane runs as a Kubernetes service in a dedicated security node. When deployed in clustered mode, CN-Series next-generation firewalls take advantage of the native auto scaling capabilities of Kubernetes to ensure security in even the most dynamic Kubernetes environments. Clustered deployments are best suited for large Kubernetes environments where a distributed deployment would be resource-intensive and cost-prohibitive.

[38] "Kubernetes Security Best Practices," Twistlock, June 6, 2019.

Figure 2-23

The CN-Series deploys natively as control and dataplane pods within the Kubernetes environment.



Native integration with Kubernetes enables CN-Series next-generation firewalls to leverage contextual information about the containers in the environment in the formulation of security policies. For instance, container namespaces can be used to define a traffic source in a firewall policy.

5G-ready firewalls (K2-Series)

5G creates disruptive business opportunities for mobile network operators because it can move beyond delivering connectivity and use security as a business enabler and competitive advantage. The evolution to 5G opens the door to exciting new services, but it also increases the number of potential intrusion points, amplifying the security impact. To tap into the 5G business opportunities with minimal risk of being exploited by bad actors, you need complete visibility and automated security across all network locations.

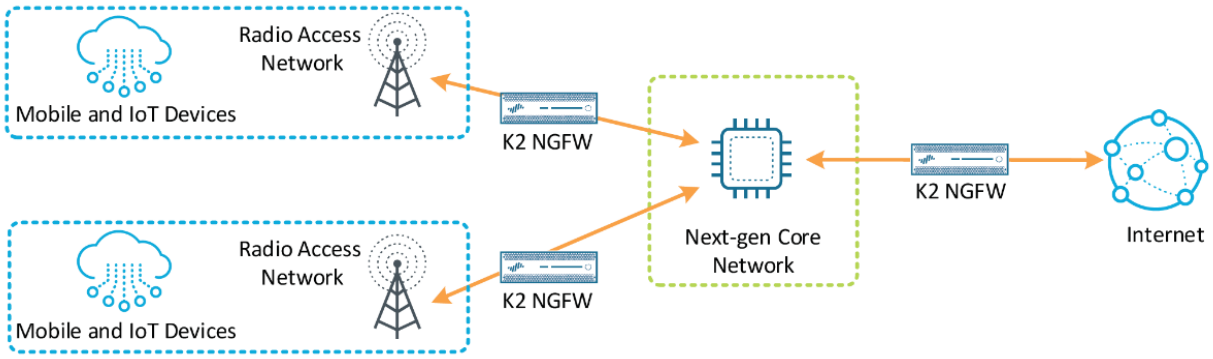
Palo Alto Networks has developed, as part of the next-generation firewall platform, a 5G-ready platform, called the K2-Series, to prevent successful cyberattacks from targeting mobile network services. The K2-Series firewalls are designed to handle growing throughput needs due to the increase of application-, user-, and device-generated data. The K2-Series offers amazing performance and threat prevention capabilities to stop advanced cyberattacks and secure mobile network infrastructure, subscribers, and services.

You can deploy K2-Series firewalls on all 5G network interfaces in order to achieve scalable, complete protection with consistent management and full application visibility (see Figure 2-24). The fundamental shift in 5G network architectures further intensifies the impact on the security landscape, with growth in the number of intrusion points, including attacks inside mobile tunnels and threats within apps traversing cellular traffic. Mobile operators need

consistent security enforcement across all network locations and all signaling traffic. This larger attack surface increases the need for application-aware Layer 7 security to detect known and unknown threats.

Figure 2-24

Securing 4G and 5G New Radio (NR) networks



K2-Series offers two modes: secure mode and express mode. Secure mode comes with all of the next-generation firewall features enabled, including threat prevention with the following enabled: App-ID, IPS, antivirus, antispyware, advanced malware analysis, and logging. Express mode is optimized for the highest throughput configuration; it is upgradable to secure mode.

2.6.1.7 IronSkillet

IronSkillet is a set of day-one, next-generation firewall configuration templates for PAN-OS that are based on security best practice recommendations.

Instead of extensive how-to documentation, the templates provide an easy-to-implement configuration model that is use case agnostic. The emphasis is on key security elements – such as dynamic updates, security profiles, rules, and logging – that should be consistent across deployments.

Palo Alto Networks has expertise in security prevention as well as in its own product portfolio. Best practice documentation is designed to provide knowledge sharing of this expertise to customers and partners. This sharing helps improve security posture across various scenarios.

The templates play a complementary role by taking common best practices recommendations and compiling them into prebuilt day-one configurations that can be readily loaded into Panorama or a next-generation firewall. The benefits include:

- Faster time to implement
- Fewer configuration errors

- Improved security posture

The templates are available on GitHub and are specific to each PAN-OS software version.

2.6.1.8 Palo Alto Networks Expedition (migration tool)

The migration to a Palo Alto Networks next-generation firewall is a critical step toward the prevention and detection of cyberattacks. Today's advanced threats require moving away from port-based firewall policies, which are no longer adequate to protect against a modern threat landscape, into an architecture that reduces your attack surface by safely enabling only those applications that are critical to your organization and eliminating applications that introduce risk.

Expedition enables organizations to analyze their existing environment, convert existing security policies to Palo Alto Networks next-generation firewalls, and assist with the transition from proof-of-concept to production.

The primary functions of Expedition include:

- **Third-party migration** transfers the various firewall rules, addresses, and service objects to a PAN-OS XML configuration file that can be imported into a Palo Alto Networks next-generation firewall. Third-party migration from the following firewall vendors is available:
    - Cisco ASA/PIX/FWSM
    - Check Point
    - Fortinet
    - McAfee Sidewinder
    - Juniper SRX/NETSCREEN
- **Adoption of App-ID** enables organizations to get the most value from their next-generation firewall, while reducing the attack surface and regaining visibility and control over the organization through App-ID.
- **Optimization** keeps next-generation firewalls operating at peak performance with services that include:
    - Architecture review
    - System health check

    - Configuration audit
    - Optional product tuning and configuration change implementation
- **Consolidation** of legacy firewalls to Palo Alto Networks virtual systems enables organizations to customize administration, networking, and security policies for the network traffic that is associated with specific departments or customers. In a standard virtual system interface configuration, each virtual system uses a dedicated interface to the internet, requiring the use of multiple IP addresses. A shared gateway allows organizations to create a common virtual interface for the virtual systems that correspond to a single physical interface. This shared gateway is helpful in environments where the ISP provides only a single IP address. All of the virtual systems communicate with the outside world through the physical interface, using a single IP address.
- **Centralized management with Panorama** enables organizations to centrally manage the process of configuring devices, deploying security policies, performing forensic analyses, and generating reports across the organization's entire network of Palo Alto Networks next-generation firewalls. Panorama and the individual device management interfaces are available as either a virtual appliance or a dedicated management platform and share the same web-based look and feel, which ensures workflow consistency while minimizing any learning curve or delay in executing tasks.
- **Auto-zoning** automatically adapts security policies from vendors that currently do not use zones and zones-based rules. The mapping of zones depends on the routes and the zone interface IP address. The mappings adjust when you set or change the interfaces and zones settings.
- **Customized response pages** can be loaded by administrators to notify end users of policy violations.

With its combination of tools, expertise, and best practices, Palo Alto Networks helps analyze an organization's existing environment and migrate policies and firewall settings to the next-generation firewall, while assisting in all phases of the transition.

2.6.1.9 Best Practice Assessment (BPA)

Most organizations don't fully implement the capabilities of their next-generation firewalls, leading to gaps in security. The Palo Alto Networks BPA is a free tool used to quickly identify the most critical security controls for an organization to focus on.

The BPA consists of the following three parts:

- The **Best Practice Assessment** is a focused evaluation of your adoption of security configuration best practices for Next-Generation Firewalls or Panorama network security management, grouped by policies, objects, networks, and devices.
- The **Security Policy Capability Adoption Heatmap** shows gaps in your capability adoption, displaying your current adoption percentage rating for each metric as well as a comparison against industry averages (see Figure 2-25). With deep insight into how you are leveraging prevention capabilities, you can continuously improve your security.
- The **BPA Executive Summary** is designed for management and executives to better understand the current state of security capability adoption at a glance—including information on progress from prior reports, if available—to help your organization confidently progress toward best practice implementation.

Figure 2-25

Security Capability Adoption Heatmap

2.6.2 Subscription services

In order for your next-generation firewall to gain complete visibility and apply full threat prevention on your network, you must activate the licenses for each of the subscription services:

- DNS Security
- URL Filtering
- Threat Prevention
- WildFire

Additional subscription services available for your next-generation firewall include:

- **IoT Security:** The IoT Security solution works with next-generation firewalls to dynamically discover and maintain a real-time inventory of the IoT devices on your network. Through AI and machine-learning algorithms, the IoT Security solution achieves a high level of accuracy, even classifying IoT device types encountered for the first time. And because it's dynamic, your IoT device inventory is always up to date. IoT Security also provides the automatic generation of policy recommendations to control IoT device traffic, as well as the automatic creation of IoT device attributes for use in firewall policies.

- **SD-WAN:** Provides intelligent and dynamic path selection on top of the industry-leading security that PAN-OS software already delivers. Managed by Panorama, the SD-WAN implementation includes centralized configuration management, automatic VPN topology creation, traffic distribution, and monitoring and troubleshooting.

- **Advanced Threat Prevention:** In addition to all of the features included with Threat Prevention, the Advanced Threat Prevention subscription provides an inline cloud-based threat detection and prevention engine, leveraging deep learning models trained on high fidelity threat intelligence gathered by Palo Alto Networks, to defend your network from evasive and unknown command-and-control (C2) threats by inspecting all network traffic.

- **Advanced URL Filtering:** Advanced URL Filtering uses a cloud-based ML-powered web security engine to perform ML-based inspection of web traffic in real-time. This reduces reliance on URL databases and out-of-band web crawling to detect and prevent advanced, file-less web-based attacks including targeted phishing, web-delivered malware and exploits, command-and-control, social engineering, and other types of web attacks.

- **Cortex Data Lake** (discussed in Section 4.4.4): Provides cloud-based, centralized log storage and aggregation. The Cortex Data Lake is required or highly-recommended to support several other cloud-delivered services, including Cortex XDR, IoT Security, and Prisma Access.

- **GlobalProtect Gateway:** Provides mobility solutions and/or large-scale VPN capabilities. By default, you can deploy GlobalProtect portals and gateways (without HIP checks) without a license. If you want to use advanced GlobalProtect features (HIP checks and related content updates, the GlobalProtect Mobile App, IPv6 connections, or a

GlobalProtect Clientless VPN) you will need a GlobalProtect Gateway license for each gateway.

- **Virtual Systems:** This is a perpetual license, and is required to enable support for multiple virtual systems on PA-3200 Series firewalls. In addition, you must purchase a Virtual Systems license if you want to increase the number of virtual systems beyond the base number provided by default on PA-5200 Series, and PA-7000 Series firewalls (the base number varies by platform). The PA-800 Series, PA-220, and VM-Series firewalls do not support virtual systems.

- **Enterprise Data Loss Prevention (DLP):** Provides cloud-based protection against unauthorized access, misuse, extraction, and sharing of sensitive information. Enterprise DLP provides a single engine for accurate detection and consistent policy enforcement for sensitive data at rest and in motion using machine learning-based data classification, hundreds of data patterns using regular expressions or keywords, and data profiles using Boolean logic to scan for collective types of data.

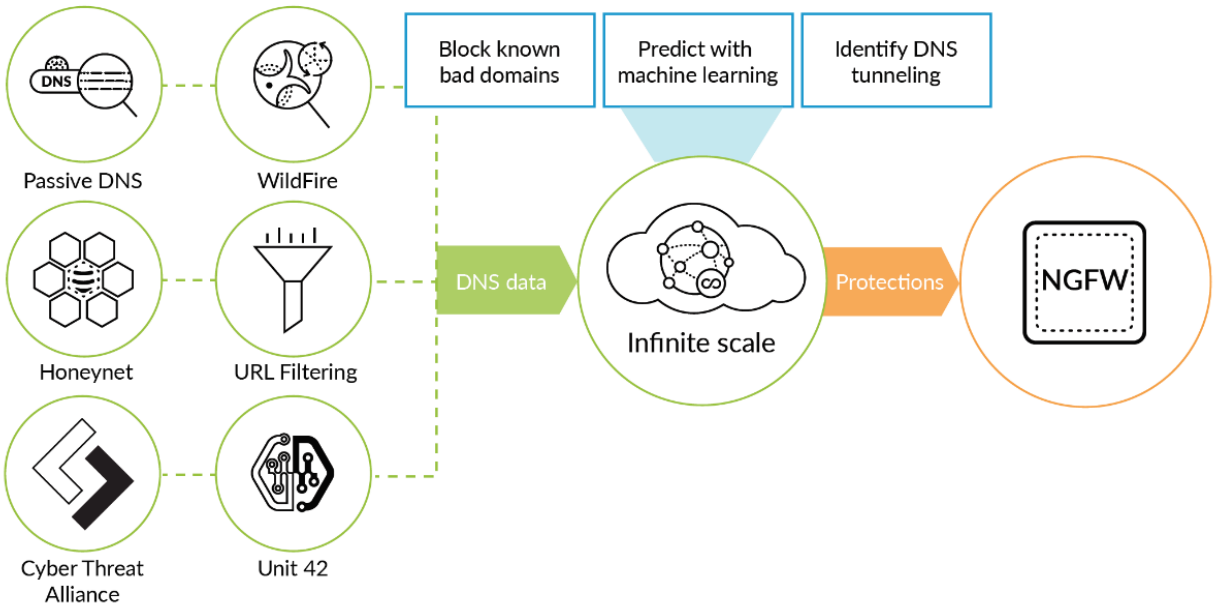
- **SaaS Security Inline:** The SaaS Security solution works with Cortex Data Lake to discover all of the SaaS applications in use on your network. SaaS Security Inline can discover thousands of Shadow IT applications and their users and usage details. SaaS Security Inline also enforces SaaS policy rule recommendations seamlessly across your existing Palo Alto Networks firewalls. App-ID Cloud Engine (ACE) also requires SaaS Security Inline.

2.6.2.1 DNS Security service

The Palo Alto Networks DNS Security service applies predictive analytics to disrupt attacks that use DNS for C2 or data theft. Tight integration with Palo Alto Networks next-generation firewalls gives you automated protection and eliminates the need for independent tools. Threats hidden in DNS traffic are rapidly identified with shared threat intelligence and machine learning. Cloud-based protections scale infinitely and are always up to date, giving your organization a critical new control point to stop attacks that use DNS (see Figure 2-26).

Figure 2-26

Rich DNS data powers machine learning for protection.



Predict and block new malicious domains

DNS is a massive and often overlooked attack surface present in every organization. Adversaries take advantage of the ubiquitous nature of DNS to abuse it at multiple points of an attack, including reliable C2. Security teams struggle to keep up with new malicious domains and enforce consistent protections for millions of emerging domains at once.

The DNS Security service takes a different approach to predicting and blocking malicious domains, giving the advantage back to overwhelmed network defenders.

Next-generation firewalls protect you against tens of millions of malicious domains identified with real-time analysis and continuously growing global threat intelligence. Your protection continues to grow with data from a large, expanding threat intelligence-sharing community. The Palo Alto Networks malicious domain database has been gathered over years, with sources including:

- WildFire malware prevention service to find new C2 domains, file download source domains, and domains in malicious email links
- URL Filtering to continuously crawl newfound or uncategorized sites for threat indicators

- Passive DNS and device telemetry to understand domain resolution history seen from thousands of deployed next-generation firewalls, generating petabytes of data per day
- Unit 42 threat research to provide human-driven adversary tracking and malware reverse engineering, including insight from globally deployed honeypots
- More than 30 third-party sources of threat intelligence to enrich understanding

With the DNS Security service, your next-generation firewalls can predict and stop malicious domains from domain generation algorithm-based malware with instant enforcement. Malware's use of domain generation algorithms (DGAs) continues to grow, limiting the effectiveness of blocking known malicious domains alone. DGA malware uses a list of randomly generated domains for C2, which can overwhelm the signature capability of traditional security approaches. DNS Security deals with DGA malware by using:

- **Machine learning** to detect new and never-before-seen DGA domains by analyzing DNS queries as they are performed
- **Easy-to-set policy** for dynamic action to block DGA domains or sinkhole DNS queries
- **Threat attribution and context** to identify the malware family with machine learning for faster investigation efforts

A cloud-based database scales infinitely to provide limitless protection against malicious domains. Your protections are always up to date, whether 10,000 or 100 million new malicious domains are created in a single day. As part of the cloud-based service, all DNS queries are checked against the Palo Alto Networks infinitely scalable, cloud-based database in real time to determine appropriate enforcement action. The DNS Security service removes one of the most effective and widely used methods by which attackers establish C2, and its protection scales infinitely, ensuring your next-generation firewalls can get ahead of new malicious domains before any harm is done.

Neutralize DNS tunneling

Advanced attackers use DNS tunneling to hide data theft or C2 in standard DNS traffic. The sheer volume of DNS traffic often means defenders simply lack the visibility or resources to universally inspect it for threats. The DNS Security service enables you to:

- Use machine learning to quickly detect C2 or data theft hidden in DNS tunneling. With historical and real-time shared threat intelligence, Palo Alto Networks algorithms observe the features of DNS queries, including query rate and patterns, entropy, and *n*-gram frequency analysis of the domains to accurately detect tunneling behavior.

- Extend PAN-OS signature-based protection to identify advanced tunneling attempts. DNS Security expands the native ability of next-generation firewalls to detect and prevent DNS tunneling. Protections are scalable and evasion-resistant, covering known and unknown variants of DNS tunneling.
- Rapidly neutralize DNS tunneling with automated policy action. DNS tunneling is automatically stopped with the combination of easy-to-set policy actions on the next-generation firewall and blocking the parent domain for all customers.

Simplify security with automation and replace standalone tools

Security teams need integrated innovations that extend the value of their existing security investments without complicating operations. DNS Security takes advantage of the next-generation firewall to stop attacks using DNS, with full automation to reduce manual effort.

Tight integration with the next-generation firewall provides a critical new control point to stop attacks that use DNS. The service ensures that you have one device to deploy, with a single set of policies to manage. Alerts are coordinated across your entire security stack, including firewall policy violations, IDS/IPS, web security, and malware analysis.

When attacks using DNS are identified, security administrators can automate the process of sinkholing malicious domains on the firewall to cut off C2 and rapidly identify infected users on the network. Combining malicious domain sinkholing, DAGs, and logging actions automates detection and response workflows, saving analysts time by removing slow and manual processes.

The DNS Security service is built on a modular, cloud-based architecture to seamlessly add new detection, prevention, and analytics capabilities with zero impact to production next-generation firewalls.

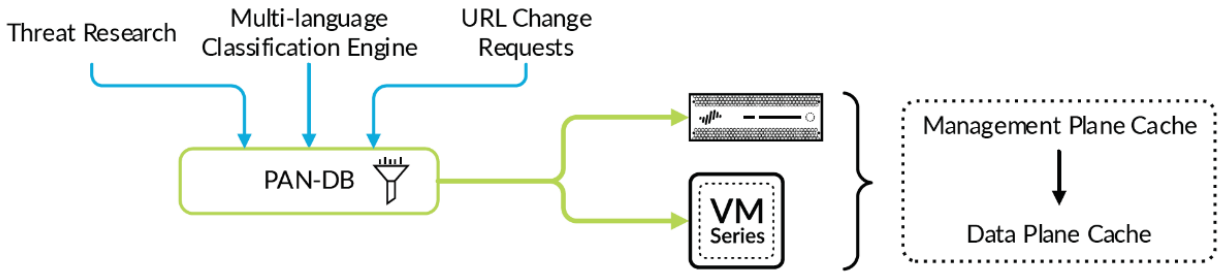
2.6.2.2 URL Filtering service

To complement the threat prevention and application control capabilities, a fully integrated, on-box URL filtering database enables security teams to not only control end-user web surfing activities but also to combine URL context with application and user rules. The URL Filtering service complements App-ID by enabling you to configure the next-generation firewall to identify and control access to websites and to protect your organization from websites hosting malware and phishing pages. You can use the URL category as a match criterion in policies, which permits exception-based behavior and granular policy enforcement. For example, you can deny access to malware and hacking sites for all users but allow access to users who belong to the IT security group.

When you enable URL Filtering, all web traffic is compared against the URL Filtering database, PAN-DB, which contains millions of URLs that have been grouped into approximately 65 categories. The malware and phishing URL categories in PAN-DB are updated in real time, which can enforce subsequent attempts to access the site based on the URL category, instead of treating it as unknown. User-credential detection, a part of URL Filtering, allows you to alert on or block users from submitting credentials to untrusted sites. If corporate credentials are compromised, user-credential detection allows you to identify who submitted credentials so that you can remediate (see Figure 2-27).

Figure 2-27

URL Filtering service



The on-box URL database can be augmented to suit the traffic patterns of the local user community with a custom URL database. For fast and easy access to frequently visited URLs, PAN-DB provides high-performance local caching, and URLs that are not categorized by the local URL database can be pulled into cache from a hosted URL database. In addition to database customization, administrators can create unique URL categories to further customize the URL controls to suit their specific needs.

URL categorization can be combined with application and user classification to further target and define policies. For example, SSL decryption can be invoked for select high-risk URL categories to ensure that threats are exposed, and QoS controls can be applied to streaming media sites. URL filtering visibility and policy controls can be bound to specific users through transparent integration with enterprise directory services (such as Active Directory, LDAP, and eDirectory), with additional insight provided through customizable reporting and logging.

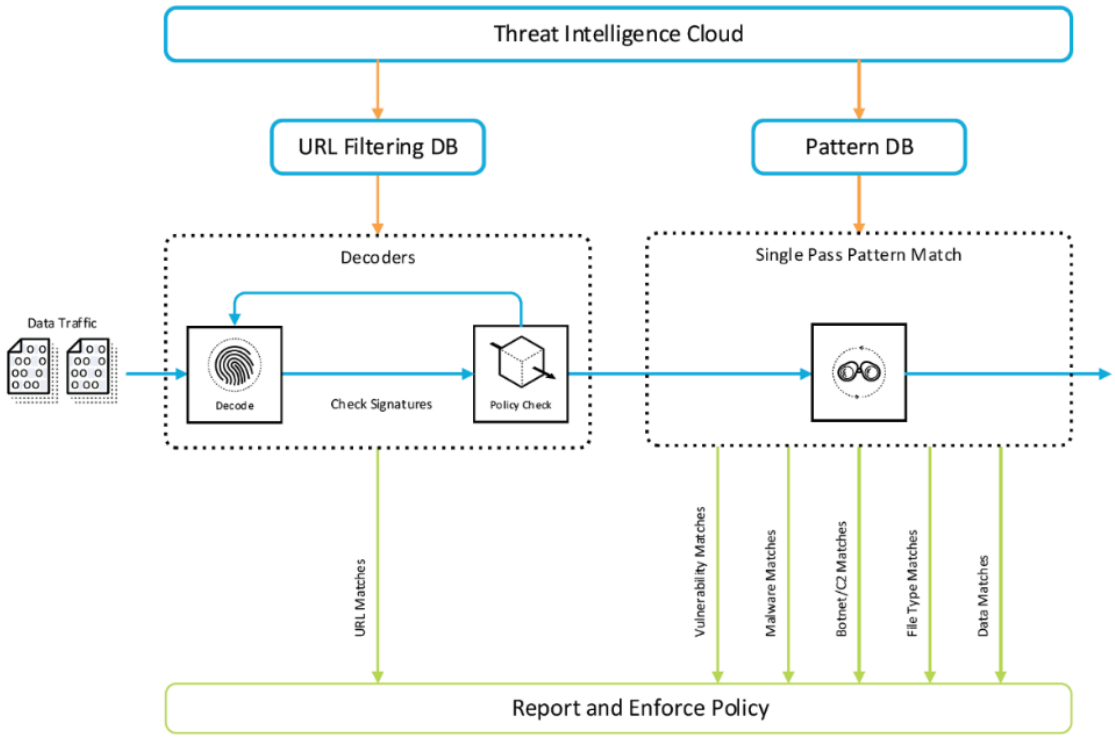
Administrators can configure a custom block page to notify end users of any policy violations. The page can include references to the username, the IP address, the URL they are attempting to access, and the URL category. To place some of the web activity ownership back in the user's hands, administrators can allow users to continue to the website or webpage after being presented with a warning page, or they can use passwords to override the URL filtering policy.

2.6.2.3 Threat Prevention service (antivirus, anti-spyware, and vulnerability protection)

Threat Prevention blocks known malware, exploits, and C2 activity on the network. Adding the Threat Prevention subscription brings additional capabilities to your next-generation firewall that identify and prevent known threats hidden within allowed applications. The Threat Prevention subscription includes malware/antivirus, C2, and vulnerability protection (see Figure 2-28).

Figure 2-28

Threat Prevention service



Malware/antivirus protection

Using content-based signatures, inline malware protection blocks malware before it ever reaches the target host. Signatures based on content detect patterns in the body of the file that identify future variations of the files, even when the content is modified slightly. This ability allows the next-generation firewall to identify and block polymorphic malware that otherwise would be treated as a new unknown file.

The stream-based scanning engine protects the network without introducing significant latency, which is a serious drawback of network antivirus offerings that rely on proxy-based scanning engines. The stream-based malware scanning inspects traffic when the first packets of the file

are received, eliminating threats as well as performance issues typical of traditional standalone solutions. Key anti-malware capabilities include:

- Inline, stream-based detection and prevention of malware hidden in compressed files and web content
- Protection against payloads hidden in common file types, such as Microsoft Office documents and PDF files

Command-and-control (spyware) protection

There are no silver bullets when it comes to preventing all threats from entering the network. After the initial infection, attackers communicate with the compromised device through a C2 channel, using it to pull down additional malware, issue further instructions, and steal data. C2 protections focus on those unauthorized communication channels and cut them off by blocking outbound requests to malicious domains and from known C2 toolkits installed on infected devices.

The C2 protection provides sinkhole capabilities for outbound requests to malicious domains, accurately identifying the compromised device and preventing data exfiltration. You can configure the sinkhole so that any outbound request to a malicious domain or IP address is redirected to one of your network's internal IP addresses. This policy effectively blocks C2 communication, preventing those requests from ever leaving the network. A report of the hosts on your network making such requests is compiled even though those hosts sit behind the DNS server. You have a daily list of potentially compromised devices on which to act, without the added stress of remediation crunch time, because communications with the attacker have already been severed.

Vulnerability protection

The Next-Generation Firewall's vulnerability protection and intrusion prevention capabilities detect and block exploit attempts and evasive techniques at both the network and the application layers. These exploits can include port scans, buffer overflows, remote code execution, protocol fragmentation, and obfuscation. Vulnerability protections are based on signature matching and anomaly detection, which decode and analyze protocols and use the learned information to block malicious traffic patterns and provide visibility through alerts. Stateful pattern matching detects attacks across multiple packets, considering arrival order and sequence – ensuring that all allowed traffic is well-intentioned and devoid of evasion techniques.

Protocol decoder-based analysis decodes the protocol and then intelligently applies signatures to detect network and application exploits. Because there are many ways to exploit a single vulnerability, the intrusion prevention signatures are based on the vulnerability itself, providing more thorough protection against a wide variety of exploits. A single signature can stop multiple exploits of a known system or application vulnerability. Protocol anomaly-based protection detects non-Request for Comments (RFC) compliant protocol use, such as an overlong uniform resource identifier (URI) or FTP login. Finally, easy-to-configure, custom vulnerability signatures allow you to tailor intrusion prevention capabilities to your network's unique needs.

2.6.2.4 Zero-day malware prevention (WildFire)

The WildFire cloud-based malware analysis environment is a cyber threat prevention service that identifies unknown malware, zero-day exploits, and advanced persistent threats (APTs) through static and dynamic analysis in a scalable, virtual environment. WildFire automatically disseminates updated protections in near real time to immediately prevent threats from spreading – all without manual intervention. Although basic WildFire support is included as part of the Threat Prevention license, the WildFire subscription service provides enhanced services for organizations that require immediate coverage for threats, frequent WildFire signature updates, advanced file type forwarding (APK, PDF, Microsoft Office, and Java Applet), as well as the ability to upload files by using the WildFire API.

As part of the next-generation firewall's inline threat prevention capability, the firewall performs a hash calculation for each unknown file, and the hash is submitted to WildFire. If any WildFire subscriber has seen the file before, then the existing verdict for that file is immediately returned. Links from inspected emails are also submitted to WildFire for analysis. Possible verdicts include:

- **Benign.** Safe and does not exhibit malicious behavior
- **Grayware.** No security risk but might display intrusive behavior (for example, adware, spyware, and browser helper objects)
- **Malware.** Malicious in nature and intent and can pose security threat (for example, viruses, worms, Trojans, rootkits, botnets, and remote-access toolkits)
- **Phishing.** Malicious attempt to trick the recipient into revealing sensitive data

If WildFire has never seen the file, the next-generation firewall is instructed to submit the file for analysis. If the file size is under the configured size limit, the next-generation firewall securely transmits the file to WildFire. Next-generation firewalls with an active WildFire license

perform scheduled auto-updates to their WildFire signatures, with update checks configured as often as every minute.

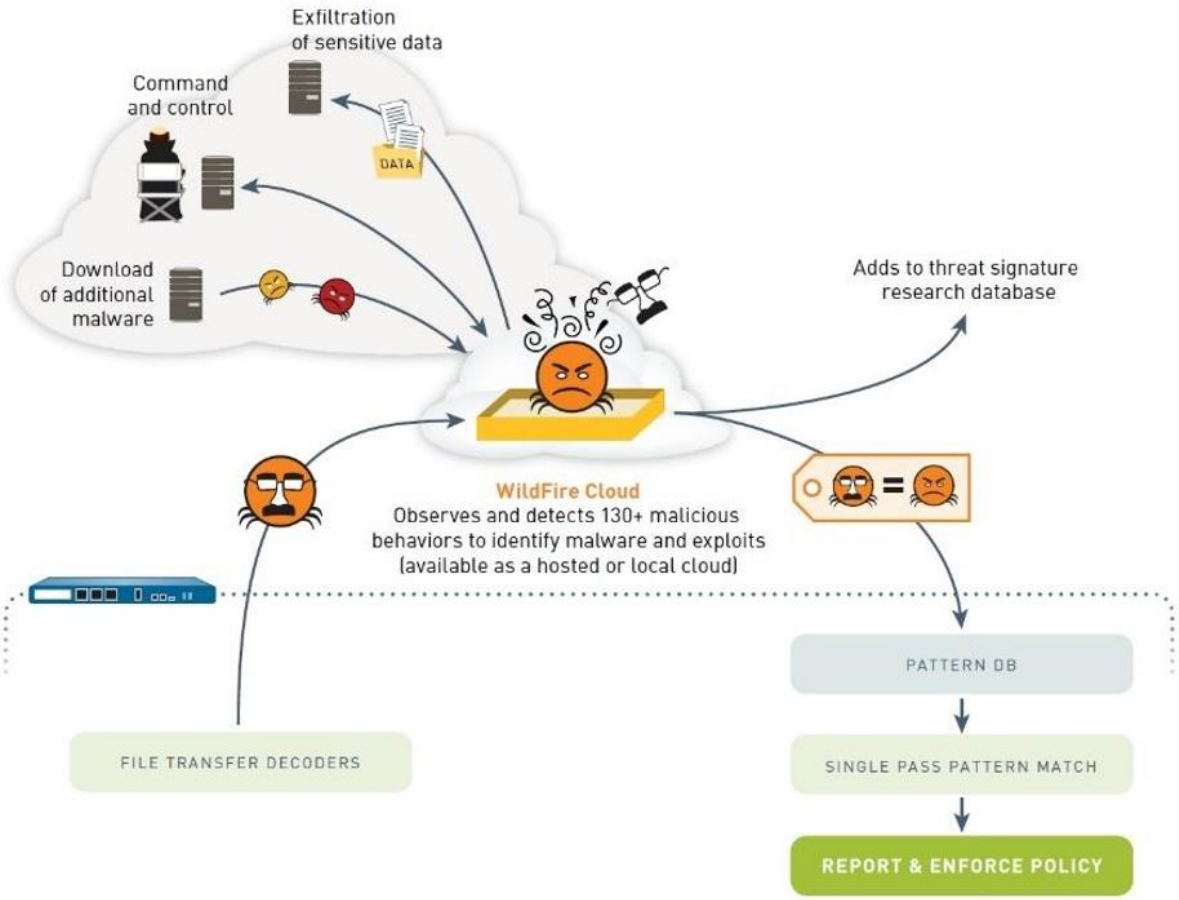
WildFire leverages inline machine learning based malware and phishing prevention (real-time WildFire verdict and anti-malware dynamic classification) to determine if the corresponding webpages for email links submitted to the service host any exploits, malware, or phishing capabilities. The behaviors and properties of the website are taken into consideration when making a verdict on the link.

WildFire significantly improves security posture and protection against unknown malware. WildFire processes about 5 million unique files daily and about 30,000 to 50,000 unique malware files that are sent to WildFire by customer-deployed Palo Alto Networks next-generation firewalls. Typically, 60 percent of these malware files are not detected by any of the major antivirus vendors when first submitted to WildFire, and 30 days later 25 to 50 percent are still not detected by the major antivirus vendors.

To support dynamic malware analysis across the network at scale, WildFire is built on a cloud-based architecture (see Figure 2-29). Where regulatory or privacy requirements prevent the use of public cloud infrastructure, a private cloud solution can be built in an on-premises data center.

Figure 2-29

WildFire provides cloud-based malware analysis and threat prevention.



In addition to leveraging either public cloud or private cloud deployments, organizations can use both within the same environment. The hybrid cloud capabilities of WildFire allow security teams more file analysis flexibility because the teams can define which file types are sent to the WildFire public cloud versus the on-premises appliance or private cloud. The WildFire hybrid cloud capability enables organizations to alleviate privacy or regulatory concerns by using the WildFire appliance for file types containing sensitive data. Organizations also benefit from the comprehensive analysis and global threat intelligence services of the WildFire public cloud for all others.

The Security Operating Platform proactively blocks known threats, which provides baseline defenses against known exploits, malware, malicious URLs, and C2 activity. When new threats emerge, the Security Operating Platform automatically routes suspicious files and URLs to WildFire for deep analysis.

WildFire inspects millions of samples per week from its global network of customers and threat intelligence partners looking for new forms of previously unknown malware, exploits, malicious

domains, and outbound C2 activity. The cloud-based service automatically creates new protections that can block targeted and unknown malware, exploits, and outbound C2 activity by observing their actual behavior rather than by relying on pre-existing signatures. The protections are delivered globally within minutes, and the result is a closed-loop, automated approach to preventing cyberthreats. This includes:

- Positive security controls to reduce the attack surface
- Inspection of all traffic, ports, and protocols to block all known threats
- Rapid detection of unknown threats by observing the actions of malware in a cloud-based execution environment
- Automatic deployment of new protections back to the frontline to ensure that threats are known to all and blocked across the attack lifecycle

Behavior-based cyberthreat discovery

To find unknown malware and exploits, WildFire executes suspicious content in the Windows, Android, and macOS operating systems, with full visibility into common file types, including:

- Executables (EXEs), dynamic-link libraries (DLLs), compressed files (ZIP), and Portable Document Format (PDF)
- Microsoft Office documents, spreadsheets, and presentations
- Java files
- Android application packages (APKs)
- Adobe Flash applets and web pages (including high-risk embedded content, such as Java and Adobe Flash files/images)

WildFire identifies hundreds of potentially malicious behaviors to uncover the true nature of malicious files based on their actions, including:

- **Changes made to host.** WildFire monitors all processes for modifications to the host, including file and registry activity, code injection, memory heap spraying (exploits), *mutexes*, Windows service activity, the addition of auto-run programs, and other potentially suspicious activities.

- **Suspicious network traffic.** WildFire performs analysis of all network activity produced by the suspicious file, including backdoor creation, downloading of next-stage malware, visiting low-reputation domains, network reconnaissance, and more.
- **Anti-analysis detection.** WildFire monitors techniques used by advanced malware that are designed to avoid virtual machine-based analysis, such as debugger detection, hypervisor detection, code injection into trusted processes, disabling of host-based security features, and more.

Key Terms

A *mutex* is a program object that allows multiple program threads to share the same resource, such as file access, but not simultaneously.

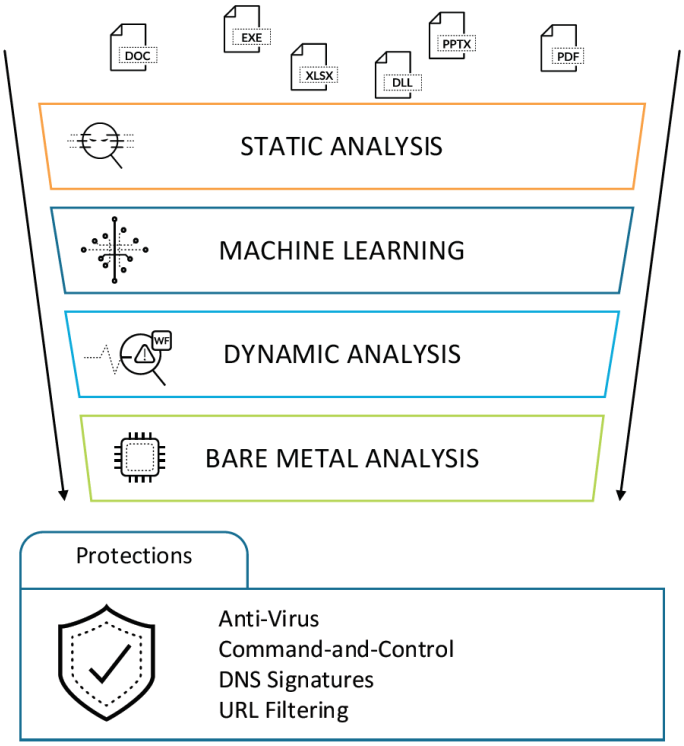
WildFire is natively integrated with the Security Operating Platform, which includes Cortex XDR (discussed in Section 4.4.1) endpoint protection and Prisma SaaS (discussed in Section 3.5.3), and it can classify all traffic across hundreds of applications. WildFire uniquely applies this behavioral analysis to web traffic, email protocols (SMTP, IMAP, and POP3), and FTP, regardless of ports or encryption.

WildFire applies the following analysis methods to submitted files (see Figure 2-30):

- **Machine learning/static analysis.** Identification of variants of known threats by comparing malware feature sets against a dynamically updated classification system. Detection of known threats by analyzing the characteristics of samples before execution.
- **Dynamic analysis.** A custom-built, evasion-resistant virtual environment in which previously unknown submissions are executed within a virtualized test environment to determine real-world effects and behavior.
- **Bare-metal dynamic analysis.** Fully hardware-based analysis environment specifically designed for advanced VM-aware threats. Samples that display the characteristics of an advanced VM-aware threat are steered toward the bare-metal appliance by the heuristic engine.

Figure 2-30

WildFire analysis



The dynamic updates from the Threat Intelligence Cloud coordinate threat prevention across the platform and are key to the prevention capabilities it provides. The unknown-threat handling methodology essentially turns unknown threats into known threats.

In addition to protecting you from malicious and exploitative files and links, WildFire looks deeply into malicious outbound communication, disrupting command-and-control (C2) activity with anti-C2 signatures and DNS-based callback signatures. WildFire also feeds this information into URL Filtering with PAN-DB, which automatically blocks newly discovered malicious URLs. This correlation of threat data and automated protections is key to identifying and blocking ongoing intrusion attempts and future attacks on your organization, without requiring policy updates and configuration commits.

Furthermore, Palo Alto Networks promotes information sharing and industry advocacy by contributing structured intelligence derived from its Threat Intelligence Cloud to the Cyber Threat Alliance (CTA). Co-founded by Palo Alto Networks and other industry leaders, the CTA is an organization working to improve the cybersecurity of the global digital ecosystem by enabling near real-time, high-quality cyberthreat information sharing within the cybersecurity community. CTA and its members share timely, actionable, contextualized, and campaign-based

intelligence that they can use to improve their products and services in order to better protect their customers, more systematically thwart adversaries, and improve the security of the digital ecosystem.

Threat prevention with global intelligence sharing

When an unknown threat is discovered, WildFire automatically generates protections to block it across the Cyberattack Lifecycle, and it shares these updates with all global subscribers within as little as five minutes. These quick updates can stop rapidly spreading malware. Additionally, these updates are payload-based, so they can block proliferation of future variants without any additional action or analysis.

WildFire protects organizations from malicious and exploitative files and links, and it also looks deep into malicious outbound communication and disrupts C2 activity with anti-C2 signatures and DNS-based callback signatures. The information is also used for URL Filtering with PAN-DB, where newly discovered malicious URLs are automatically blocked. This correlation of threat data and automated protections is key to identifying and blocking ongoing intrusion attempts and future attacks against your organization.

Integrated logging, reporting, and forensics

WildFire provides access to integrated logs, analysis, and visibility into WildFire events, through the management interface, the WildFire portal, and Panorama (discussed in Section 2.6.3). This access enables security teams to quickly investigate and correlate events observed in their networks to rapidly locate the data needed for timely investigations and incident response.

Host-based and network-based *indicators of compromise* (IoCs) become actionable through log analysis and custom signatures. To aid security and incident response teams in discovering infected hosts, WildFire also provides:

- Detailed analysis of every malicious file sent to WildFire across multiple operating system environments, including host-based and network-based activity.
- Session data associated with the delivery of the malicious file, including source, destination, application, User-ID, and URL.
- Access to the original malware sample for reverse engineering and full packet captures (pcaps) of dynamic analysis sessions.
- An open application programming interface (API) for integration with best-in-class security information and event management (SIEM) tools (such as the Palo Alto

Networks application for Splunk), and leading endpoint agents. This analysis provides numerous IoCs that can be applied across the attack lifecycle.

- Native integration with Cortex XDR endpoint protection (discussed in Section 4.4.1) and Prisma SaaS (discussed in Section 3.5.3).
- Access to the actionable intelligence and global context provided by CORTEX SOAR TIM (discussed in Section 4.4.3).
- Native integration with the correlation engine in Palo Alto Networks next-generation firewalls (discussed in Section 2.6.1).

Key Terms

An *indicator of compromise* (IoC) is a network or operating system (OS) artifact that provides a high level of confidence that a computer security incident has occurred.

A *packet capture* (pcap) is a traffic intercept of data packets that can be used for analysis.

2.6.3 Network security management (Panorama)

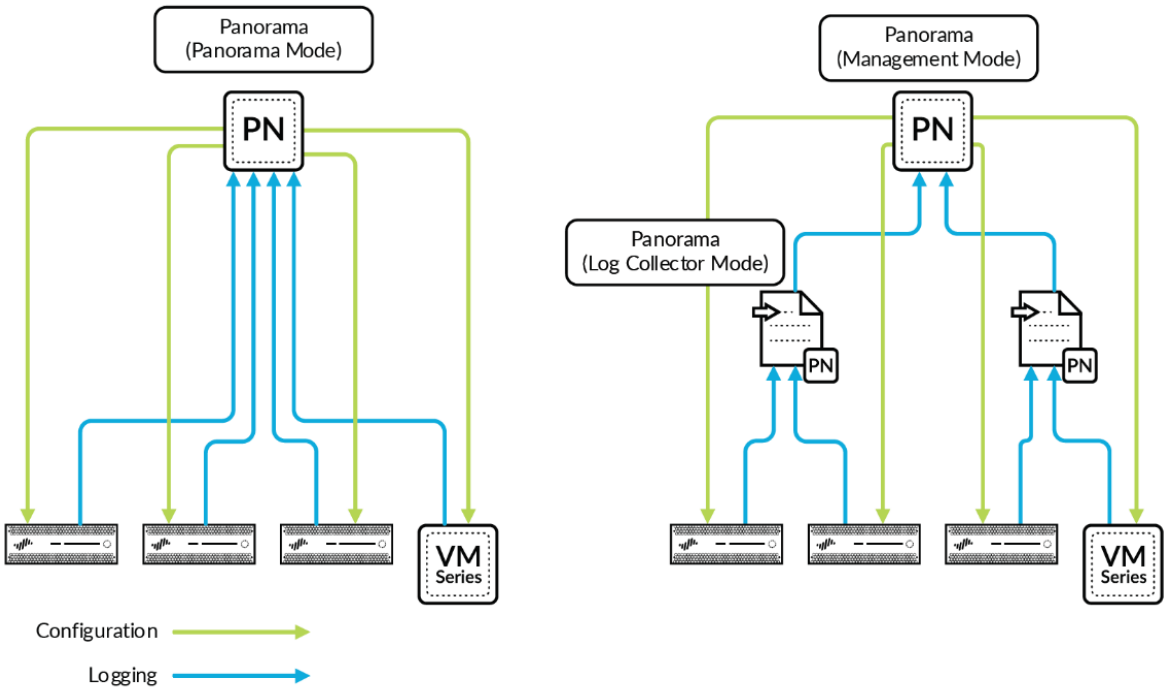
Panorama enables you to manage all key features of the Palo Alto Networks next-generation firewalls by using a model that provides central oversight and local control. You can deploy Panorama as either an on-premises hardware appliance or a virtual appliance, but you can also deploy it as a virtual appliance in the public cloud.

Three deployment mode options are available for Panorama, which (if necessary) allows for the separation of management and log collection (see Figure 2-31):

- **Panorama mode.** Panorama controls both policy and log management functions for all the managed devices.
- **Management Only mode.** Panorama manages configurations for the managed devices but does not collect or manage logs.
- **Log Collector mode.** One or more Log Collectors collect and manage logs from the managed devices. This assumes that another deployment of Panorama is operating in Management Only mode.

Figure 2-31

Panorama deployment modes



The separation of management and log collection enables the Panorama deployment to meet scalability, organizational, and geographical requirements. The choice of form factor and deployment mode gives you the maximum flexibility for managing Palo Alto Networks next-generation firewalls in a distributed network.

Panorama reduces security management complexity with consolidated policy creation and centralized management features. The Application Command Center (ACC) in Panorama provides a customizable dashboard for the setup and control of Palo Alto Networks next-generation firewalls with an efficient rulebase and actionable insight into network-wide traffic and threats.

Panorama simplifies network security management with a single security rulebase for firewall, threat prevention, URL filtering, application awareness, user identification, sandboxing, file blocking, and data filtering to safely enable applications in the enterprise. Security rules can be easily imported, duplicated, or modified across the network. Centralized management of policies and objects provides consistent global security for the organization, and local administrative control provides flexibility at the local level.

The time it takes to deploy changes across dozens or hundreds of firewalls can be costly in the number of employees required and the delay that projects experience while they wait for the

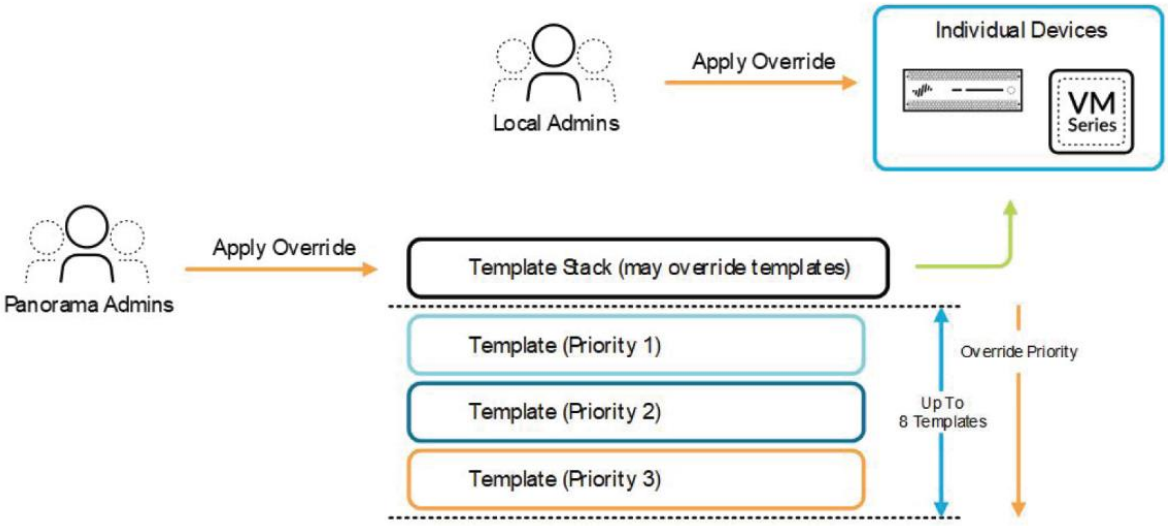
process to be completed. In addition to time, errors can increase when network and security engineers program changes firewall by firewall. Panorama provides the following tools for centralized administration that can reduce time and errors for your firewall management operation:

- **Templates/template stacks.** Panorama manages common device and network configuration through templates. You can use templates to manage configuration centrally and then push the changes to all managed firewalls. This approach avoids making the same individual firewall change repeatedly across many devices. Templates are grouped together within a template stack, and the stack is applied to selected firewalls.

  You can define common building blocks for device and network configuration within a template. These building blocks are logically combined by adding them to a template stack. If there are no overlapping parameters, then the stack reflects the combination of all the individual templates. If there is overlap, then the settings from the highest priority template take precedence. You can override the template settings at the stack level. A local administrator can also perform overrides directly on an individual device if necessary (see Figure 2-32).

Figure 2-32

Panorama template stack and templates



Firewall-specific settings such as IP addresses must be unique per device. Instead of using overrides, these settings can be managed by using variables within templates. Panorama manages the variable assignments at deployment time, either per device

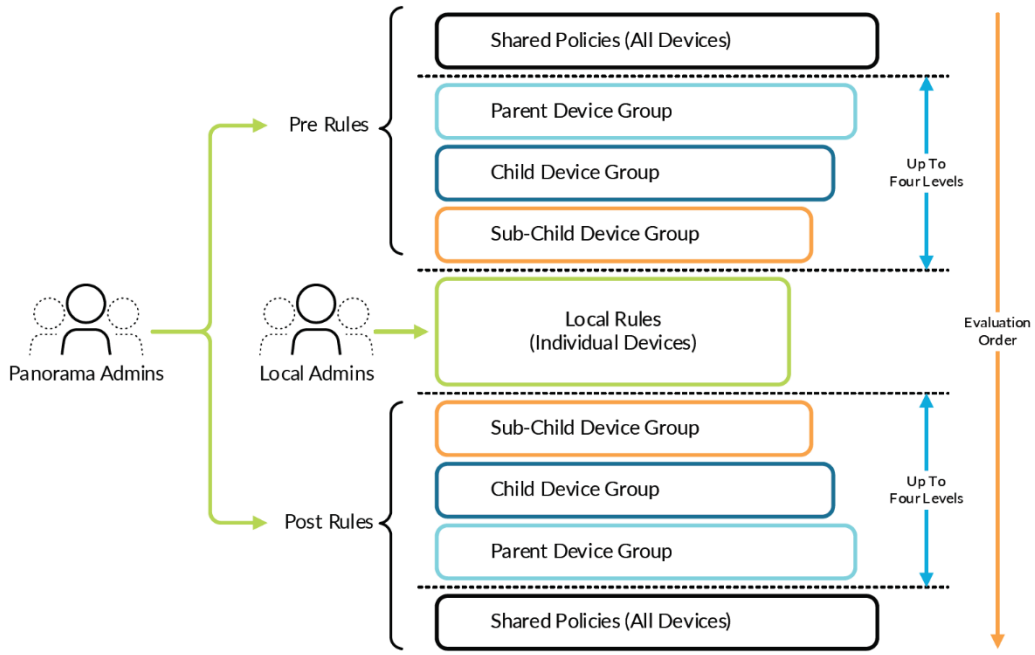
through manual assignment or in bulk by importing a spreadsheet with the settings for multiple devices.

- **Hierarchical device groups.** Panorama manages common policies and objects through hierarchical device groups. It uses multilevel device groups to centrally manage the policies across all deployment locations with common requirements. For example, device groups may be determined geographically, such as Europe and North America. Also, each device group can have a functional subdevice group (for example, perimeter or data center).

  You can define shared policies for central control while granting your local firewall administrator the autonomy to make specific local adjustments. At the device group level, you can create common policies that are defined as the first set of rules ("pre rules") and the last set of rules ("post rules") to be evaluated against match criteria. Pre rules and post rules can be viewed on a managed firewall, but they can only be edited in Panorama within the context of the defined administrative roles. Local device rules (those between pre rules and post rules) can be edited by either your local firewall administrator or a Panorama administrator who has switched to a local firewall context. In addition, you can reference shared objects defined by a Panorama administrator in locally managed device rules (see Figure 2-33).

Figure 2-33

Panorama device groups and policy evaluation



Role-based administration delegates feature-level access, including availability of data (enabled, read-only, or disabled and hidden from view), to different members of your staff. You can give specific individuals access to tasks that are pertinent to their job while making other tasks either hidden or read-only.

As your deployment grows in size, you can make sure updates are sent to downstream boxes in an organized manner. For instance, you may prefer to centrally qualify a software update before it is delivered via Panorama to all production firewalls at once. Using Panorama, you can centrally manage the update process for software updates, content application updates, antivirus signatures, threat signatures, URL filtering database, and licenses.

Panorama can also integrate with your IT workflow applications. When a log is generated on the next-generation firewall, Panorama can trigger actions and initiate workflows through HTTP-based APIs. Selective log forwarding allows you to define the criteria to automate a workflow or an action. Although you can integrate with any HTTP-based service that exposes an API, predefined formatting for ServiceNow and VMware NSX Manager allow you to create incident reports and to tag virtual machines.

Panorama uses the same set of powerful monitoring and reporting tools available at the local device management level. As you perform log queries and generate reports, Panorama dynamically pulls the most current data directly from managed next-generation firewalls or from logs forwarded to Panorama. Logging and reporting capabilities in Panorama include:

- **Log viewer.** For either an individual device or all devices, you can quickly view log activities using dynamic log filtering by clicking a cell value and/or using the expression builder to define the sort criteria. Results can be saved for future queries or exported for further analysis.

- **Custom reporting.** Predefined reports can be used as-is, customized, or grouped together as one report to suit specific requirements.

- **User activity reports.** A user activity report shows the applications used, URL categories visited, websites visited, and all URLs visited over a specified period of time for individual users. Panorama builds the reports using an aggregate view of users' activity, no matter which firewall they are protected by or which IP address or device they may be using.

- **Log forwarding.** Panorama aggregates logs collected from all of your Palo Alto Networks firewalls, both physical and virtual form factor, and forwards them to a remote destination for purposes such as long-term storage, forensics, or compliance reporting. Panorama can forward all or selected logs, Simple Network Management Protocol

(SNMP) traps, and email notifications to a remote logging destination, such as a syslog server (over UDP, TCP, or SSL).

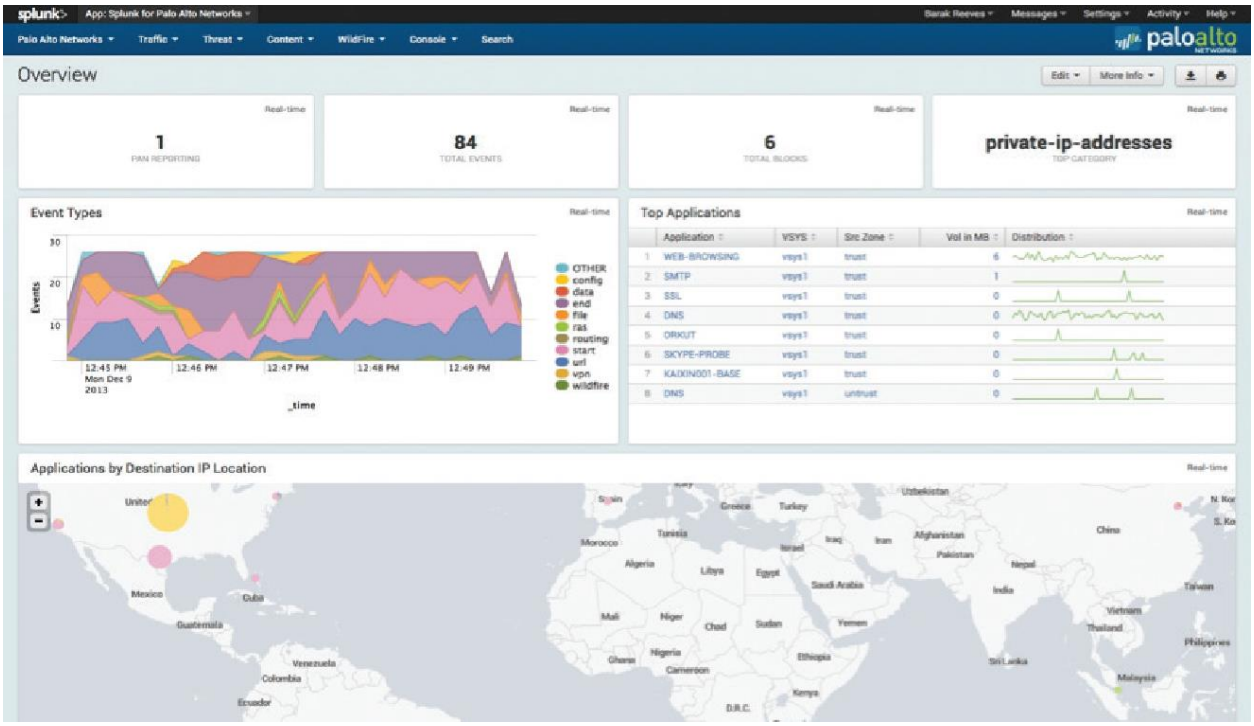
Panorama can be deployed in a centralized architecture with all Panorama management and logging functions consolidated into a single device, or in a distributed architecture with separate management units and Log Collectors in a hierarchical deployment architecture. This includes:

- **Panorama manager.** The Panorama manager is responsible for handling the tasks associated with policy and device configuration across all managed devices. The manager does not store log data locally but rather uses separate Log Collectors for handling log data. The manager analyzes the data stored in the Log Collectors for centralized reporting.
- **Panorama Log Collector.** Organizations with high logging volume and retention requirements can deploy dedicated Panorama Log Collector devices that will aggregate log information from multiple managed firewalls.

Palo Alto Networks and Splunk have partnered to extend the powerful visibility into network traffic from Panorama to other network components. The combined solution delivers highly effective, coordinated detection, incident investigation, and response for cyberthreats. With the Splunk App for Palo Alto Networks (see Figure 2-34), enterprise security teams have a powerful platform for security visualization, monitoring, and analysis that enables them to fully leverage the extensive application, user, content, and threat data generated by Palo Alto Networks devices.

Figure 2-34

Integration with Splunk extends visibility and prevention capabilities to your entire network infrastructure.



The integrated solution not only combines several approaches for identifying cyberthreats – including dynamic sandbox analysis, statistical anomaly detection, and infrastructure-wide event correlation – but also enables security administrators to expedite incident response by automating the steps needed to block malicious sources and quarantine compromised devices.

2.6.4 Wi-Fi security (Okyo Garde)

Many enterprises quickly shifted to a work-from-home (WFH) model during the global pandemic, treating WFH as an extension of mobile users "working from anywhere", to enable their workforces to continue being productive while working remotely. In many organizations, WFH has now become the "new normal", requiring enterprises to treat employee home networks more like a "branch of one". Securing and managing hundreds and thousands of home office branches requires a new security, networking, and manageability paradigm.

Okyo Garde extends the corporate Wi-Fi network and security policies into employees' homes. It provides employees at home the digital experiences they had in the office and provides the security teams with the visibility, control, and cloud-scale manageability needed to extend the corporate network to the home.

Okyo Garde is an integrated security and networking solution, which complements Prisma Access (discussed in Section 3.4.2) to deliver a secure campus-like experience to WFH employees. Prisma Access is an agent-based cloud-delivered security solution that secures employee devices equipped with an endpoint agent to securely connect to the VPN to work from home.

2.6 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Multiple Choice.** Which option is *not* a defining characteristic of a next-generation firewall? (Choose one.)

    a) low latency packet processing with minimal throughput loss

    b) adherence to strict port and protocol enforcement for allow or block decisions

    c) integrated security tools

    d) bidirectional full-stack analysis of packets

2. **Short Answer.** List the three core capabilities of a next-generation firewall.

3. **Multiple Choice.** Which option is *not* a core technique for identifying applications in Palo Alto Networks next-generation firewalls? (Choose one.)

    a) packet headers

    b) application signatures

    c) protocol decoding

    d) behavioral analysis

4. **Short Answer.** List three methods of mapping user identification to an IP address within a next-generation firewall.

5. **Short Answer.** Describe stream-based malware scanning and explain its benefits.

6. **Short Answer.** What is the advantage of using templates in Panorama?

Module 3 – Fundamentals of Cloud Security

Knowledge Objectives

- Discuss cloud-computing service and deployment models, and the shared responsibility model.
- Describe cloud-native technologies — including virtual machines, containers, and orchestration — and serverless computing.
- Discuss cloud-native security — including Kubernetes security, DevOps, and DevSecOps — and visibility, governance, and compliance challenges.
- Discuss hybrid data-center security design concepts, including traditional data-center security solution weaknesses, east-west traffic protection, and security in hybrid data centers.
- Describe the requirements for cloud application security and the capabilities of Prisma Cloud.
- Describe the capabilities and requirements in a secure access service edge (SASE) and describe the basic functionality of Prisma Access.
- Demonstrate an understanding of unique SaaS-based security risks and how Prisma SaaS protects SaaS-based applications and data.

3.0 Cloud Computing

Cloud computing is not a location but rather a pool of resources that can be rapidly provisioned in an automated, on-demand manner. The U.S. National Institute of Standards and Technology (NIST) defines cloud computing in Special Publication (SP) 800-145 as “a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (such as networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction.”

The value of cloud computing is the ability to pool resources to achieve economies of scale and agility. Both public and private clouds have this ability to pool resources. Instead of having many independent and often underused servers deployed for your enterprise applications, pools of resources are aggregated, consolidated, and designed to be elastic enough to scale with the needs of your organization.

The move toward cloud computing not only brings cost and operational benefits but also technology benefits. Data and applications are easily accessed by users no matter where they reside, projects can scale easily, and consumption can be tracked effectively. Virtualization is a critical part of a cloud-computing architecture that, when combined with software orchestration and management tools, allows you to integrate disparate processes so that they can be automated, easily replicated, and offered on an as-needed basis.

3.0.1 Cloud Service Models

NIST defines three distinct cloud-computing service models:

- **Software as a service (SaaS).** Customers are provided access to an application running on a cloud infrastructure. The application is accessible from various client devices and interfaces, but the customer has no knowledge of, and does not manage or control, the underlying cloud infrastructure. The customer may have access to limited user-specific application settings, and security of the customer data is still the responsibility of the customer.

- **Platform as a service (PaaS).** Customers can deploy supported applications onto the provider's cloud infrastructure, but the customer has no knowledge of, and does not manage or control, the underlying cloud infrastructure. The customer has control over the deployed applications and limited configuration settings for the application-hosting environment. The company owns the deployed applications and data, and it is therefore responsible for the security of those applications and data.

- **Infrastructure as a service (IaaS).** Customers can provision processing, storage, networks, and other computing resources, and they can deploy and run operating systems and applications. However, the customer has no knowledge of, and does not manage or control, the underlying cloud infrastructure. The customer has control over operating systems, storage, and deployed applications, along with some networking components (for example, host firewalls). The company owns the deployed applications and data, and it is therefore responsible for the security of those applications and data.

3.0.2 Cloud Deployment Models

NIST also defines these four cloud-computing deployment models:

- **Public.** A cloud infrastructure that is open to use by the general public. It is owned, managed, and operated by a third party (or parties), and it exists on the cloud provider's premises.

- **Community.** A cloud infrastructure that is used exclusively by a specific group of organizations.
- **Private.** A cloud infrastructure that is used exclusively by a single organization. It may be owned, managed, and operated by the organization or a third party (or a combination of both), and it may exist on premises or off premises.
- **Hybrid.** A cloud infrastructure that comprises two or more of the aforementioned deployment models, bound by standardized or proprietary technology that enables data and application portability (for example, failover to a secondary data center for disaster recovery or content delivery networks across multiple clouds).

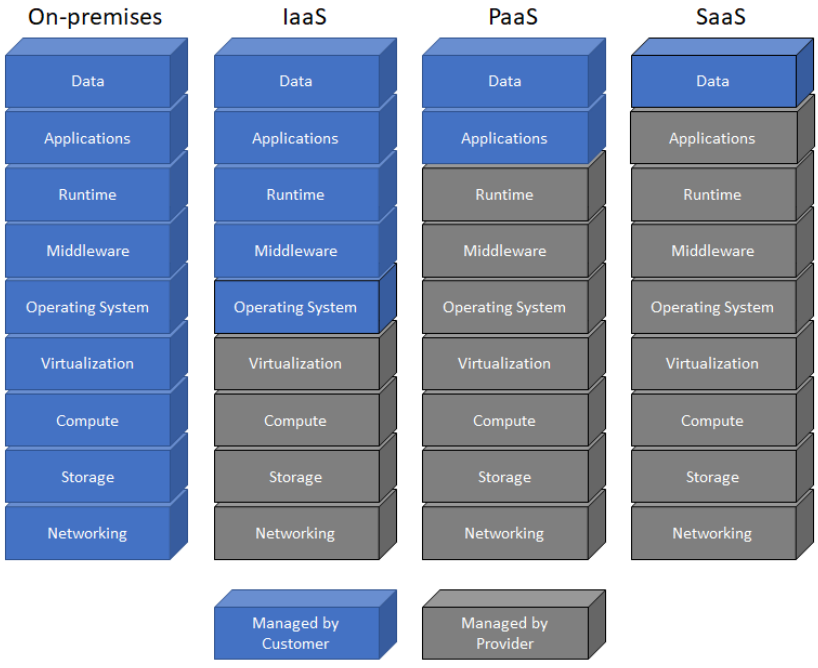
3.0.3 Cloud Security Challenges

The security risks that threaten your network today do not change when you move to the cloud. The *shared responsibility model* defines who (customer and/or provider) is responsible for what (related to security) in the public cloud.

In general terms, the cloud provider is responsible for security *of* the cloud, including the physical security of the cloud data centers, and foundational networking, storage, compute, and virtualization services. The cloud customer is responsible for security *in* the cloud, which is further delineated by the cloud service model (see Figure 3-1).

Figure 3-1

The shared responsibility model



For example, in an infrastructure-as-a-service (IaaS) model, the cloud customer is responsible for the security of the operating systems, middleware, runtime, applications, and data. In a platform-as-a-service (PaaS) model, the cloud customer is responsible for the security of the applications and data, and the cloud provider is responsible for the security of the operating systems, middleware, and runtime. In a SaaS model, the cloud customer is responsible only for the security of the data, and the cloud provider is responsible for the full stack from the physical security of the cloud data centers to the application. Multitenancy in cloud environments, particularly in SaaS models, means that customer controls and resources are necessarily limited by the cloud provider.

With the use of cloud-computing technologies, your data-center environment can evolve from a fixed environment where applications run on dedicated servers toward an environment that is dynamic and automated, where pools of computing resources are available to support application workloads that can be accessed anywhere, anytime, from any device.

Security remains a significant challenge when you embrace this new, dynamic cloud-computing fabric environment. Many of the principles that make cloud computing attractive are counter to network security best practices:

- **Security requires isolation and segmentation; the cloud relies on shared resources.** Security best practices dictate that mission-critical applications and data be isolated in secure segments on the network using the Zero Trust (discussed in Section 1.3.2) principle of "never trust, always verify." On a physical network, Zero Trust is relatively straightforward to accomplish using firewalls and policies based on application and user identity. In a cloud-computing environment, direct communication between virtual machines (VMs) within a server and in the data center (east-west traffic, discussed in Section 3.3.2) occurs constantly, in some cases across varied levels of trust, making segmentation a difficult task. Mixed levels of trust may weaken an organization's security posture when combined with virtualized port-based security offerings being unable to view intra-host traffic.

- **Security deployments are process-oriented; cloud computing environments are dynamic.** The creation or modification of your cloud workloads can often be done in minutes, yet the security configuration for this workload may take hours, days, or weeks. Security delays are not intentional; they're the result of a process that is designed to maintain a strong security posture. Policy changes need to be approved, the appropriate firewalls need to be identified, and the relevant policy updates need to be determined. In contrast, the cloud is a highly dynamic environment, with workloads (and IP addresses) constantly being added, removed, and changed. The result is a disconnect between security policy and cloud workload deployments; that disconnect

leads to a weakened security posture. Security technologies and processes must leverage capabilities such as cloning and scripted deployments to automatically scale and take advantage of the elasticity of the cloud while maintaining a strong security posture.

- **Multitenancy is a key characteristic of the public cloud – and a key risk.** Although public cloud providers strive to ensure isolation between their various customers, the infrastructure and resources in the public cloud are shared. Inherent risks in a shared environment include misconfigurations, inadequate or ineffective processes and controls, and the "noisy neighbor" problem (excessive network traffic, disk I/O, or processor utilization can negatively impact other customers sharing the same resource). In hybrid and multicloud environments that connect numerous public and/or private clouds, the lines become still more blurred, complexity increases, and security risks become more challenging to address.

- **Traditional network and host security models don't work in the cloud for serverless applications.** Historically, defense in depth was mostly performed through network layer controls. Advanced threat prevention tools are able to recognize the applications that traverse the network and determine whether they should be allowed. This type of security is still very much required in cloud-native environments, but it is no longer sufficient on its own. Public cloud providers offer a rich portfolio of services, and the only way to govern and secure many of them is through *identity and access management* (IAM). IAM controls the permissions and access for users and cloud resources. IAM policies are sets of permission policies that can be attached to either users or cloud resources to authorize what they access and what they can do with what they access.

Key Terms

Identity and access management (IAM) is a framework of business processes, policies, and technologies that facilitates the management of electronic or digital identities.

As organizations transition from a traditional data-center architecture to a public, private, or hybrid cloud environment, enterprise security strategies must be adapted to support changing requirements in the cloud. Key requirements for securing the cloud include:

- **Consistent security in physical and virtualized form factors.** The same levels of application control and threat prevention should be used to protect both your cloud-computing environment and your physical network. First, you need to be able to

confirm the identity of your applications, validating their identity and forcing them to use only their standard ports. You also need to be able to block the use of rogue applications while simultaneously looking for and blocking misconfigured applications. Finally, application-specific threat prevention policies should be applied to block both known and unknown malware from moving into and across your network and cloud environment.

- **Your business applications segmented using Zero Trust principles.** To fully maximize the use of computing resources, a relatively common current practice is to mix application workload trust levels on the same compute resource. Although they are efficient in practice, mixed levels of trust introduce new security risks in the event of a compromise. Your cloud security solution needs to be able to implement security policies based on the concept of Zero Trust (discussed in Section 1.3.2) as a means of controlling traffic between workloads while preventing lateral movement of threats.

- **Centrally managed business applications; streamlined policy updates.** Physical network security is still deployed in almost every organization, so it's critical to have the ability to manage both hardware and virtual-form-factor deployments from a centralized location using the same management infrastructure and interface. To ensure that security keeps pace with the speed of change that your workflows may exhibit, your security solution should include features that will allow you to reduce, and in some cases eliminate, the manual processes that security-policy updates often require.

No matter which type of cloud service you use, the burden of securing certain types of workloads will always fall on you, never your vendor. To maximize your cloud-environment security, consider the following best practices:

- **Review default settings.** While certain settings are automatically set by the provider, some must be manually activated. It's always better to have your own set of security policies than to assume that the vendor is taking care of a particular aspect of your cloud-native security.

- **Adapt data storage and authentication configurations to your organization.** All locations where data will be uploaded should be password protected. In addition, password-expiration policies should be carefully selected to suit the needs of your organization.

- **Don't assume your cloud data is safe.** Never assume that vendor-encrypted data is totally safe. Some vendors provide encryption services before upload, and some do not.

Whichever the case, make sure to encrypt your data in transit and at rest by using your own keys.

- **Integrate with your cloud's data-retention policy.** Understanding your vendor's data-retention and -deletion policy is essential. It's important to have multiple copies of your data and to have a fixed data-retention period. But what happens when you delete data from the cloud? Is it still accessible to the vendor? Are there other places where it might have been cached or copied? You should verify these things up front when setting up a new cloud environment.

- **Set appropriate privileges.** Appropriate settings for privilege levels go a long way toward making your cloud environment more secure. By using role-based access controls (RBACs) for authorization, you can ensure that every person who views or works with your data has access only to the things that are absolutely necessary.

- **Keep cloud software up to date.** Your vendor may provide infrastructure and, in some cases, a prebuilt software environment or cloud-native firewall. But anything that you add is your responsibility to secure. Thus, it's your responsibility as a user to ensure that your security patches, operating systems, etc. are up to date. The simplest way to prevent *technical debt* and backlogs is to automate the updates.

- **Build security policies and best practices into your cloud images.** Leaving your cloud-native security to different developers on your DevOps security team might result in policy discrepancies. A good way to combat this is to create cloud images with security tools configured and policies applied so that developers can simply create instances of them.

- **Isolate your cloud resources.** To reduce the risk of bad actors gaining complete control over your system, you should separate admin accounts for development, deployment, testing, and so on. That way, if a bad actor accesses one account, they cannot laterally move to other aspects of the environment.

Key Terms

Technical debt is a software-development concept that has also been applied more generally to IT. It refers to anticipating additional future costs of rework required because of an earlier decision or course of action that was necessary for agility but was not necessarily the most optimal or appropriate decision or course of action.

3.0 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Multiple Choice.** In which cloud-computing service model does a provider's applications run on a cloud infrastructure and the consumer does not manage or control the underlying infrastructure? (Choose one.)

    a) platform as a service (PaaS)

    b) infrastructure as a service (IaaS)

    c) software as a service (SaaS)

    d) public cloud

2. **Fill in the Blank.** A _____cloud infrastructure comprises two or more cloud deployment models, bound by standardized or proprietary technology that enables data and application portability.

3. **Fill in the Blank.** The _____ defines who (customer and/or provider) is responsible for what, related to security, in the public cloud.

3.1 Cloud-Native Technologies

Like a new universe, the cloud-native ecosystem has many technologies and projects quickly spinning off and expanding from the initial core of containers. An especially intense area of innovation is workload deployment along with management technologies. While Kubernetes has become the industry-standard general-purpose container orchestrator, other technologies like serverless add abstract complexity associated with managing hardware and operating systems. The differences between these technologies are often small and nuanced, which makes it challenging to understand the benefits and tradeoffs.

As defined by the Cloud Native Computing Foundation's (CNCF) charter, cloud-native systems have the following properties:

- **Container packaged:** Running applications and processes in software containers as isolated units of application deployment, and as mechanisms to achieve high levels of resource isolation. Improves overall developer experience, fosters code and component reuse, and simplifies operations for cloud-native applications.

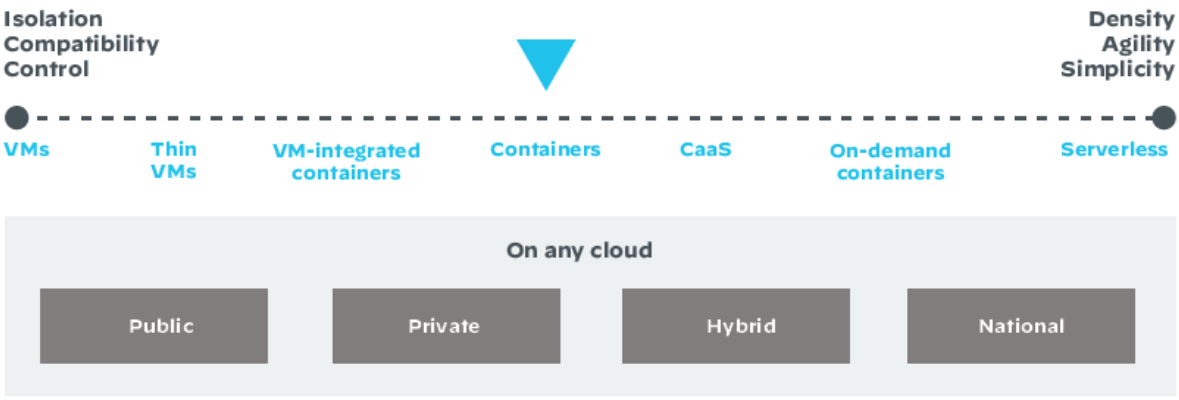
- **Dynamically managed:** Actively scheduled and actively managed by a central orchestrating process. Radically improves machine efficiency and resource utilization while reducing the cost associated with maintenance and operations.
- **Microservices oriented:** Loosely coupled with dependencies explicitly described (for example, through service endpoints). Significantly increases the overall agility and maintainability of applications. The foundation will shape the evolution of the technology to advance the state of the art for application management, and to make the technology ubiquitous and easily available through reliable interfaces.

A useful way to think of cloud-native technologies is as a continuum spanning from virtual machines (VMs) to containers to serverless. On one end are traditional VMs operated as stateful entities, as we've done for over a decade now. On the other are completely stateless, serverless apps that are effectively just bundles of app code without any packaged accompanying operating-system (OS) dependencies.

In between, there are things like Docker, the new Amazon Web Services (AWS) Fargate service, container-as-a-service (CaaS) platforms, and other technologies that try to provide a different balance between compatibility and isolation on one hand, and agility and density on the other. That balance is the reason for such diversity in the ecosystem. Each technology tries to place the fulcrum at a different point, but the ends of the spectrum are consistent: one end prioritizes familiarity and separation, while the other trades off some of those characteristics for increased abstraction and less deployment effort (see Figure 3-2).

Figure 3-2

The continuum of cloud-native technologies



There's a place for all these technologies—they are different tools with different advantages and tradeoffs, and organizations typically use at least a few of them simultaneously. That

heterogeneity is unlikely to change as organizations bring increasingly more-critical workloads into their cloud-native stacks, especially those with deep legacy roots.

3.1.1 Virtualization

Virtualization technology emulates real – or physical – computing resources, such as servers (compute), storage, networking, and applications. Virtualization allows multiple applications or server workloads to run independently on one or more physical resources.

A *hypervisor* allows multiple, virtual ("guest") operating systems to run concurrently on a single physical host computer. The hypervisor functions between the computer operating system and the hardware kernel. There are two types of hypervisors:

- **Type 1** (*native* or *bare-metal*). Runs directly on the host computer's hardware
- **Type 2** (*hosted*). Runs within an operating-system environment

Virtualization is a key technology used in data centers and cloud computing to optimize resources. Important security considerations associated with virtualization include:

- **Dormant virtual machines (VMs).** In many data-center and cloud environments, inactive VMs are routinely (often automatically) shut down when they are not in use. VMs that are shut down for extended periods of time (weeks or months) may be inadvertently missed when anti-malware updates and security patches are applied.
- **Hypervisor vulnerabilities.** In addition to vulnerabilities within the hosted applications, VMs, and other resources in a virtual environment, the hypervisor itself may be vulnerable, which can expose hosted resources to attack.
- **Intra-VM communications.** Network traffic between virtual hosts, particularly on a single physical server, may not traverse a physical switch. This lack of visibility increases troubleshooting complexity and can increase security risks because of inadequate monitoring and logging capabilities.
- **VM sprawl.** Virtual environments can grow quickly, leading to a breakdown in change-management processes and exacerbating security issues such as dormant VMs, hypervisor vulnerabilities, and intra-VM communications.

Key Terms

A *hypervisor* allows multiple, virtual (or guest) operating systems to run concurrently on a single physical host computer.

A *native* (also known as a *Type 1* or *bare-metal*) hypervisor runs directly on the host computer's hardware.

A *hosted* (also known as a *Type 2*) hypervisor runs within an operating-system environment.

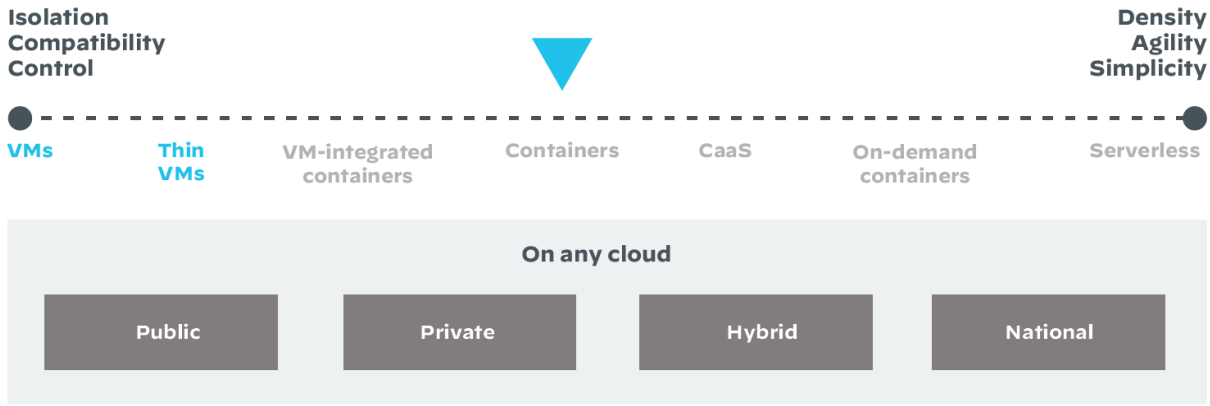
3.1.1.1 Virtual machines

While it may be surprising to see VMs discussed in the context of cloud native, the reality is that the vast majority of the world's workloads today run "directly" (non-containerized) in VMs. Most organizations do not see VMs as a legacy platform to eliminate, nor simply as a dumb host on which to run containers. Rather, they acknowledge that many of their apps have not yet been containerized and that the traditional VM is still a critical deployment model for them. While a VM not hosting containers doesn't meet all three attributes of a cloud-native system, it nevertheless can be operated dynamically and run microservices.

VMs provide the greatest levels of isolation, compatibility, and control in the continuum and are suitable for running nearly any type of workload. Examples of VM technologies include VMware vSphere, Microsoft Hyper-V, and the instances provided by almost every IaaS cloud provider, such as Amazon EC2. VMs are differentiated from "thin VMs" to their right on the continuum (see Figure 3-3) because they're often operated in a stateful manner with little separation between OS, app, and data.

Figure 3-3

VMs and thin VMs on the continuum of cloud-native technologies



3.1.1.2 Thin virtual machines

Less a distinct technology than a different operating methodology, "thin" VMs are typically the same underlying technology as VMs but deployed and run in a much less stateful manner. Thin VMs are typically deployed through automation with no human involvement, are operated as fleets rather than individual entities, and prioritize separation of OS, app, and data. Whereas a VM may store app data on the OS volume, a thin VM would store all data on a separate volume that could easily be reattached to another instance. While thin VMs also lack the container attribute of a cloud-native system, they typically have a stronger emphasis on dynamic management than traditional VMs. Whereas a VM may be set up and configured by a human operator, a thin VM would typically be deployed from a standard image, using automation tools like Puppet, Chef, or Ansible, with no human involvement.

Thin VMs are differentiated from VMs to their left on the continuum (see Figure 3-3) by the intentional focus on data separation, automation, and disposability of any given instance. They are differentiated from VM-integrated containers to their right on the continuum by a lack of a container runtime. Thin VMs have apps that are installed directly on their OS file system and executed directly by the host OS kernel without any intermediary runtime.

3.1.2 Containers and orchestration

Developers have widely embraced containers because they make building and deploying cloud-native applications simpler than ever. Not only do containers eliminate much of the friction typically associated with moving application code from testing through to production, application code packaged up as containers can also run anywhere. All the dependencies associated with any application are included within the containerized application. That makes a

containerized application highly portable across virtual machines or bare-metal servers running in a local data center or in a public cloud.

That level of flexibility enables developers to make huge gains in productivity that are too great to ignore. However, as is the case with the rise of any new IT architecture, cloud-native applications still need to be secured. Container environments bring with them a range of cybersecurity issues involving images, containers, hosts, runtimes, registries, and orchestration platforms, all of which need to be secured.

Kubernetes is an open-source orchestration platform that provides an API that enables developers to define container infrastructure in a declarative fashion – that is, infrastructure as code (IaC). Leveraging Kubernetes orchestration and a microservices architecture, organizations can publish, maintain, and update containerized cloud-native applications rapidly and at scale.

3.1.2.1 VM-integrated containers

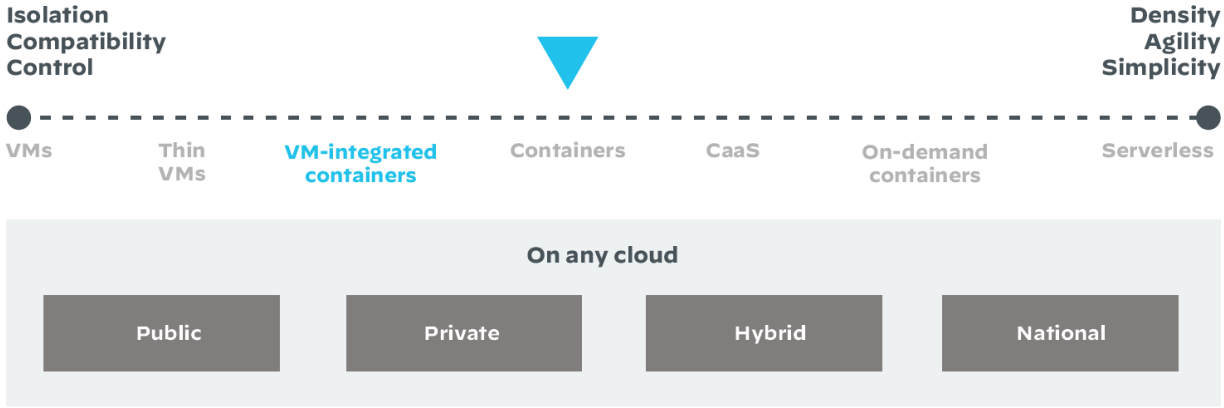
For some organizations, especially large enterprises, containers provide an attractive app-deployment and operational approach but lack sufficient isolation to mix workloads of varying sensitivity levels. Recently discovered hardware flaws like Meltdown and Spectre aside, VMs provide a much stronger degree of isolation but at the cost of increased complexity and management burden. VM-integrated containers, like Kata containers and VMware vSphere Integrated Containers, seek to accomplish this by providing a blend of a developer-friendly API and abstraction of app from the OS while hiding the underlying complexities of compatibility and security isolation within the hypervisor.

Basically, these technologies seek to provide VMs without users having to know they're VMs or manage them. Instead, users execute typical container commands like "docker run," and the underlying platform automatically and invisibly creates a new VM, starts a container runtime within it, and executes the command. The end result is that the user has started a container in a separate operating-system instance, isolated from all others by a hypervisor. These VM-integrated containers typically run a single container (or set of closely related containers akin to a pod in Kubernetes) within a single VM. VM-integrated containers possess all three cloud-native system attributes and typically don't even provide manual configuration as an optional deployment approach.

VM-integrated containers are differentiated from thin VMs to their left on the continuum (see Figure 3-4) because they are explicitly designed to solely run containers and tightly integrate VM provisioning with container runtime actions. They're differentiated from pure containers to their right on the continuum by the mapping of a single container per OS instance and the integrated workflow used to instantiate a new VM, along with the container it hosts, via a singular, container-centric flow.

Figure 3-4

VM-integrated containers on the continuum of cloud native technologies



3.1.2.2 Containers

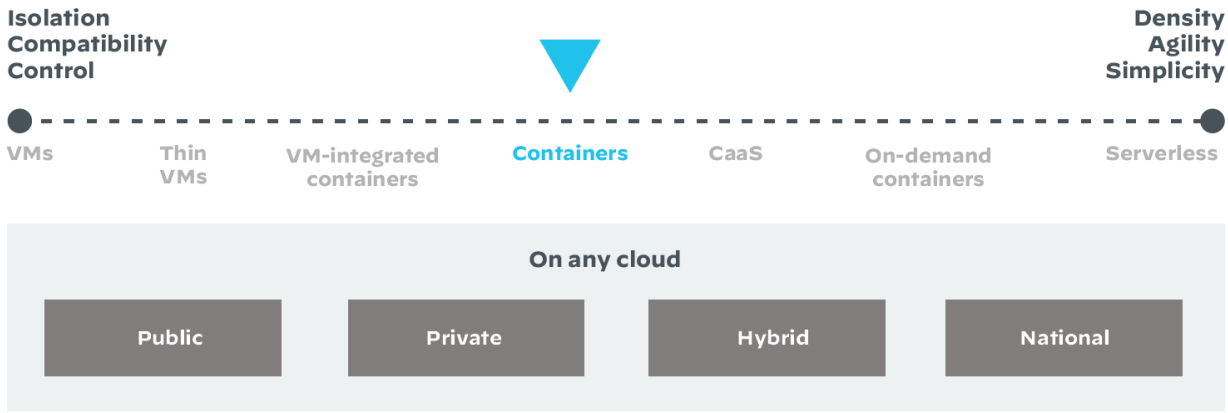
Containers deliver all three cloud-native system characteristics as well as providing a balanced set of capabilities and trade-offs across the continuum. Popularized by, and best known from, the Docker project, containers have existed in various forms for many years and have their roots in technologies like Solaris Zones and BSD Jails. While Docker is a well-known brand, other vendors are adopting its underlying technologies of `runc` and `containerd` to create similar but separate solutions.

Containers balance separation (though not as well as VMs), excellent compatibility with existing apps, and a high degree of operational control with good density potential and easy integration into software development flows. Containers can be complex to operate, primarily due to their broad configurability and the wide variety of choices they present to operational teams. Depending on these choices, containers can be either completely stateless, dynamic, and isolated; highly intermingled with the host operating system and stateful; or anywhere in between. This degree of choice is both the greatest strength and the greatest weakness of containers. In response, the market has created systems to their right on the continuum—such as serverless—to both make them easier to manage at scale and abstract some of their complexity by reducing some configurability.

Containers are differentiated from VM-integrated containers to their left on the continuum (see Figure 3-5) by neither using a strict 1:1 mapping of container to VM nor wrapping the provisioning of the underlying host operating system into the container deployment flow. They're differentiated from container-as-a-service platforms to their right on the continuum by requiring users to be responsible for deployment and operation of all the underlying infrastructure, including not just hardware and VMs but also the maintenance of the host operating systems within each VM.

Figure 3-5

Containers on the continuum of cloud-native technologies



3.1.2.3 Containers as a Service

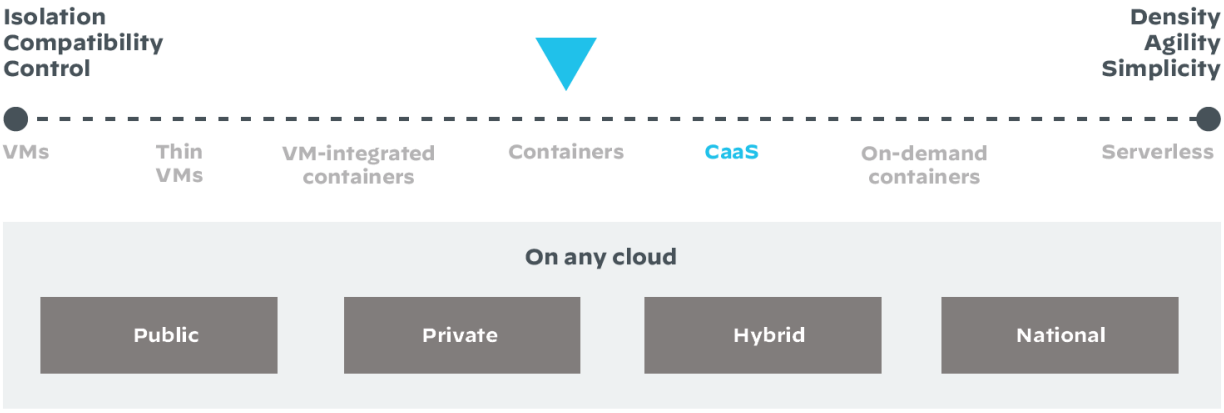
As containers grew in popularity and their uses diversified, orchestrators like Kubernetes (and its derivatives like OpenShift), Mesos, and Docker Swarm became increasingly important to deploy and operate containers at scale. While they abstract much of the complexity required to deploy and operate large numbers of microservices composed of many containers and running across many hosts, these orchestrators themselves can be complex to set up and maintain. Additionally, these orchestrators are focused on the container runtime and do little to assist with the deployment and management of underlying hosts. While sophisticated organizations often use technologies like thin VMs wrapped in automation tooling to address this, even these approaches do not fully unburden the organization from managing the underlying compute, storage, and network hardware. Container-as-a-service (CaaS) platforms provide all three cloud-native characteristics by default and, while assembled from many more generic components, are highly optimized for container workloads.

Since major public cloud IaaS providers already have extensive investments in lower-level automation and deployment, many have chosen to leverage this advantage to build complete platforms for running containers that strive to relieve users of the burden of managing the underlying hardware and VMs. These CaaS platforms include Google Kubernetes Engine, Azure Kubernetes Service, and Amazon EC2 Container Service. These solutions combine the container deployment and management capabilities of an orchestrator with their own platform-specific APIs to create and manage VMs. This integration allows users to provision capacity more easily, without the need to manage the underlying hardware or virtualization layer. Some of these platforms, such as Google Kubernetes Engine, even use thin VMs running container-focused operating systems, such as Container-Optimized OS (CoreOS), to further reduce the need to manage the host operating system.

CaaS platforms are differentiated from containers to their left on the continuum (see Figure 3-6) by providing a more comprehensive set of capabilities that abstract the complexities involved with hardware and VM provisioning. They are differentiated from on-demand containers to their right on the continuum by typically still enabling users to directly manage the underlying VMs and host OS. For example, in most CaaS deployments, users can SSH directly to a node and run arbitrary tools as a root user to aid in diagnostics or customize the host OS.

Figure 3-6

CaaS platform on the continuum of cloud-native technologies



3.1.2.4 On-demand containers

While CaaS platforms simplify the deployment and operation of containers at scale, they still provide users with the ability to manage the underlying host OS and VMs. For some organizations, this flexibility is highly desirable, but in other use cases, it can be an unneeded distraction. Especially for developers, the ability to simply run a container, without any knowledge or configuration of the underlying hosts or VMs, can increase development efficiency and agility.

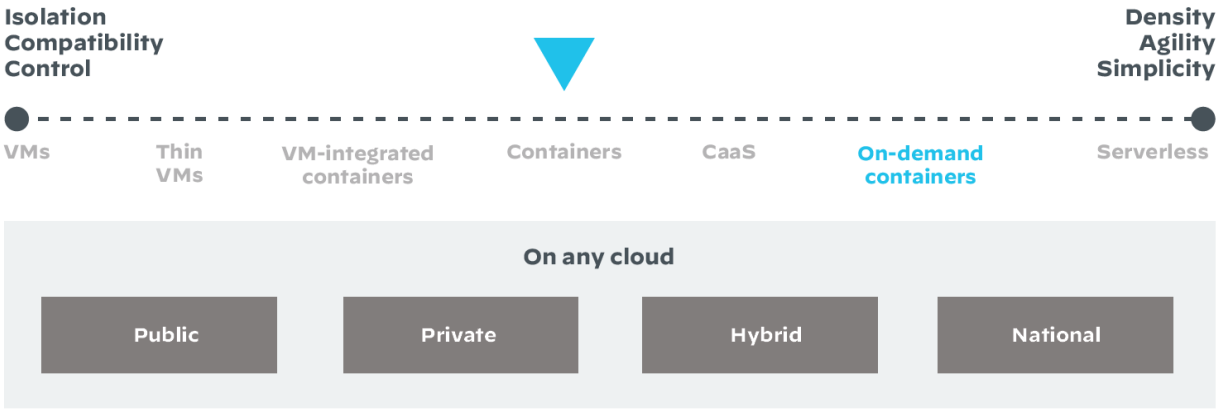
On-demand containers are a set of technologies designed to trade off some of the compatibility and control of CaaS platforms for lessened complexity and ease of deployment. On-demand container platforms include AWS Fargate and Azure Container Instances. On these platforms, users may not have any ability to directly access the host OS and must exclusively use the platform interfaces to deploy and manage their container workloads. These platforms provide all three cloud-native attributes and arguably even require them; it's typically impractical not to build apps for them as microservices, and the environment can only be managed dynamically and deployed as containers.

On-demand containers are differentiated from CaaS platforms to their left on the continuum (see Figure 3-7) by the lack of support for direct control of the host OS and VMs, along with the

requirement that typical management occurs through platform-specific interfaces. They are differentiated from serverless to their right on the continuum because on-demand containers still run normal container images that could be executed on any other container platform. For example, the same image that a user may run directly in a container on their desktop can be run unchanged on a CaaS platform or in an on-demand container. The consistency of an image format as a globally portable package for apps, including all their underlying OS-level dependencies, is a key difference from serverless environments.

Figure 3-7

On-demand containers on the continuum of cloud-native technologies



3.1.3 Serverless computing

Serverless architectures (also referred to as function as a service, or FaaS) enable organizations to build and deploy software and services without maintaining or provisioning any physical or virtual servers. Applications made using serverless architectures are suitable for a wide range of services and can scale elastically as cloud workloads grow.

From a software-development perspective, organizations adopting serverless architectures can focus on core product functionality and completely disregard the underlying operating system, application server, or software runtime environment.

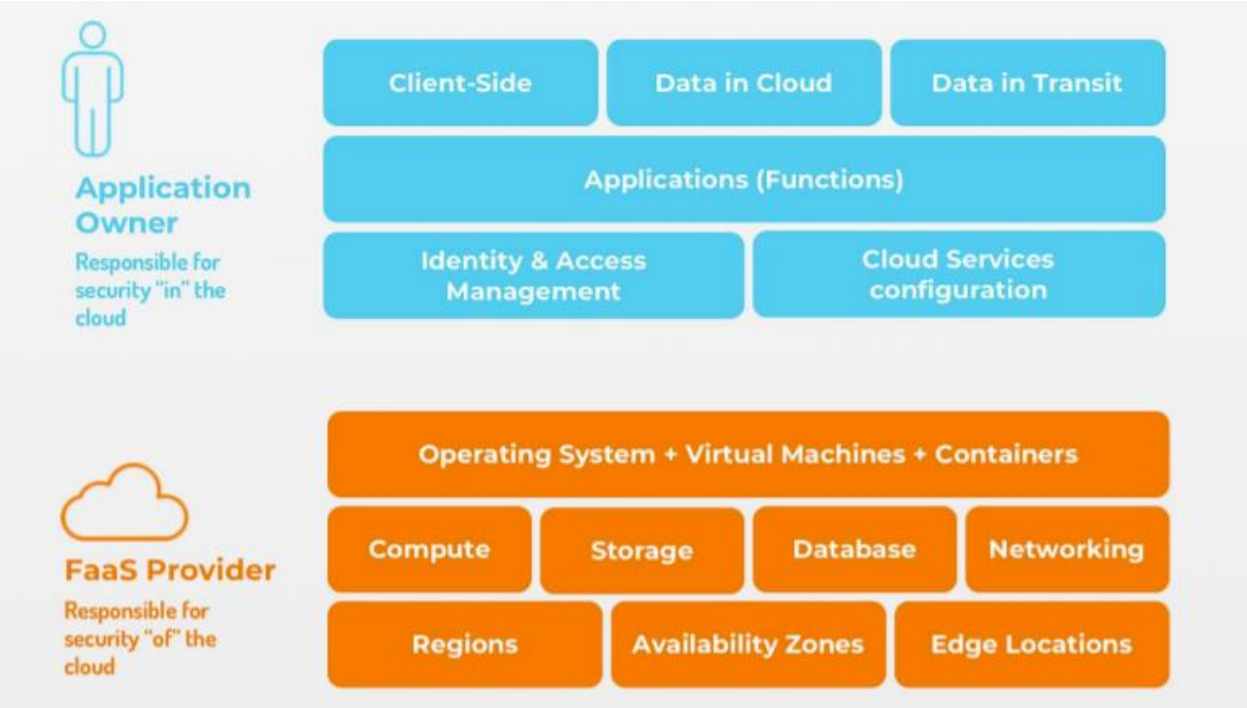
By developing applications using serverless architectures, users relieve themselves of the daunting task of continually applying security patches for the underlying operating system and application servers. Instead, these tasks are now the responsibility of the serverless architecture provider.

In serverless architectures, the serverless provider is responsible for securing the data center, network, servers, operating systems, and their configurations. However, application logic, code,

data, and application-layer configurations still need to be robust and resilient to attacks. These are the responsibility of application owners (see Figure 3-8).

Figure 3-8

Serverless architectures and the shared-responsibility model



Adopting a serverless model can impact application development in several ways:

- **Reduced operational overhead.** With no servers to manage, developers and DevOps don't need to worry about scaling infrastructure, installing and maintaining agents, or other infrastructure-related operations.

- **Increased agility.** Because serverless applications rely heavily on managed services for things like databases and authentication, developers are free to focus on the business logic of the application, which will typically runs on an FaaS, such as AWS Lambda or Google Cloud Functions.

- **Reduced costs.** With most services used in serverless applications, the customer pays only for usage. For example, with AWS Lambda, customers pay for the executions of their functions. This pricing model typically has a significant impact on cost because customers don't have to pay for unused capacity as they would with virtual machines.

While on-demand containers greatly reduce the "surface area" exposed to end users and, thus, the complexity associated with managing them, some users prefer an even simpler way to

deploy their apps. Serverless is a class of technologies designed to allow developers to provide only their app code to a service, which then instantiates the rest of the stack below it automatically.

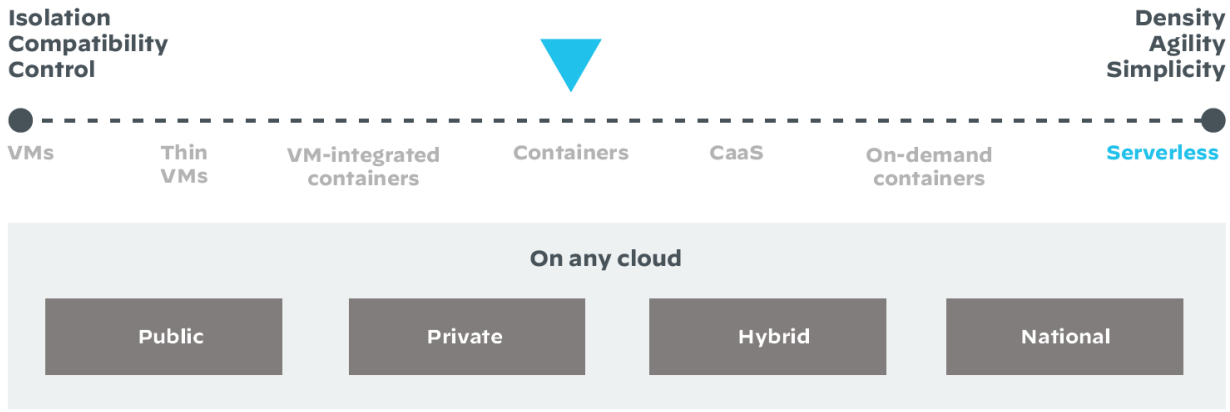
In serverless apps, the developer only uploads the app package itself, without a full container image or any OS components. The platform dynamically packages it into an image, runs the image in a container, and (if needed) instantiates the underlying host OS and VM as well as the hardware required to run them. In a serverless model, users make the most dramatic trade-offs of compatibility and control for the simplest, most efficient deployment and management experience.

Examples of serverless environments include Amazon Lambda and Azure Functions. Arguably, many PaaS offerings, such as Pivotal Cloud Foundry, are also effectively serverless even if they have not historically been marketed as such. While on the surface, serverless may appear to lack the container-specific, cloud-native attribute, containers are extensively used in the underlying implementations, even if those implementations are not exposed to end users directly.

Serverless is differentiated from on-demand containers to the left on the continuum (see Figure 3-9) by the complete inability to interact with the underlying host and container runtime, often to the extent of not even having visibility into the software that the host runs.

Figure 3-9

Serverless on the continuum of cloud-native technologies



Serverless architectures introduce a new set of issues that must be considered when securing such applications, including:

- **Increased attack surface:** Serverless functions consume data from a wide range of event sources, such as HyperText Transfer Protocol (HTTP) application program interfaces (APIs), message queues, cloud storage, Internet of Things (IoT) device communications,

and so forth. This diversity increases the potential attack surface dramatically, especially when messages use protocols and complex message structures. Many of these messages cannot be inspected by standard application-layer protections, such as web application firewalls (WAFs).

- **Attack surface complexity:** The attack surface in serverless architectures can be difficult for some to understand, given that such architectures are still somewhat new. Many software developers and architects have yet to gain enough experience with the security risks and appropriate security protections required to secure such applications.

- **Overall system complexity:** Visualizing and monitoring serverless architectures is still more complicated than standard software environments.

- **Inadequate security testing:** Performing security testing for serverless architectures is more complex than testing standard applications, especially when such applications interact with remote third-party services or with backend cloud services, such as Non-Structured Query Language (NoSQL) databases, cloud storage, or stream processing services. Additionally, automated scanning tools are currently not adapted to examining serverless applications. Common scanning tools currently include the following:

    - **Dynamic application security testing (DAST)** tools will only provide testing coverage for HTTP interfaces. This limited capability poses a problem when testing serverless applications that consume input from non-HTTP sources or interact with backend cloud services. Also, many DAST tools inadequately test web services—for example, Representational State Transfer (RESTful)—that don't follow the classic HyperText Markup Language (HTML)/HTTP request/response model and request format.

    - **Static application security testing (SAST)** tools rely on data-flow analysis, control flow, and semantic analysis to detect vulnerabilities in software. Since serverless applications contain multiple distinct functions that are stitched together using event triggers and cloud services (for example, message queues, cloud storage, or NoSQL databases), statically analyzing data flow in such scenarios is highly prone to false positives. Conversely, SAST tools will suffer from false negatives as well, since source/sink rules in many tools do not consider FaaS constructs. These rule sets will need to evolve to provide proper support for serverless applications.

    - **Interactive application security testing (IAST) tools** have better odds at accurately detecting vulnerabilities in serverless applications when compared to

both DAST and SAST. However, similar to DAST tools, their security coverage is impaired when serverless applications use non-HTTP interfaces to consume input. Furthermore, IAST solutions require that the tester deploy an instrumentation agent on the local machine, which is not an option in serverless environments.

- **Traditional security protections (firewall, web application firewall (WAF), intrusion prevention system (IPS)/intrusion detection system (IDS)):** Since organizations that use serverless architectures do not have access to the physical (or virtual) server or its operating system, they cannot deploy traditional security layers, such as endpoint protection, host-based intrusion prevention, WAFs, and so forth. Additionally, existing detection logic and rules have yet to be "translated" to support serverless environments.

3.1 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Fill in the Blank.** A _____ allows multiple virtual operating systems to run concurrently on a single physical host computer.
2. **Multiple Choice.** Which three important security considerations are associated with virtualization? (Choose three.)
    a) dormant VMs
    b) hypervisor vulnerabilities
    c) hypervisor sprawl
    d) intra-VM communications

3.2 Cloud-Native Security

The speed and flexibility that are so desirable in today's business world have led companies to adopt cloud technologies that require not just more security but new security approaches. In the cloud, you can have hundreds or even thousands of instances of an application, presenting exponentially greater opportunities for attack and data theft.

Public cloud service providers have done a great job of taking on the build, maintenance, and updating of computing hardware and providing VMs, data storage, and databases to their

customers, along with the baseline security to protect it all. But it's still up to the customer to provide security for the data, hosts, containers, and serverless instances in the cloud.

The following sections explore cloud-native security issues, including securing Kubernetes clusters, DevOps and DevSecOps, and visibility, governance, and compliance challenges.

3.2.1 The 4C's of cloud native security

The Cloud Native Computing Foundation (CNCF) Kubernetes project defines a container security model for Kubernetes in the context of cloud-native security. This model is referred to as "the 4 C's of cloud-native security." Each layer provides a security foundation for the next layer. The 4 C's of cloud-native security are:

- **Cloud** – The cloud (as well as data centers) provides the trusted computing base for a Kubernetes cluster. If the cluster is built on a foundation that is inherently vulnerable or configured with poor security controls, then the other layers cannot be properly secured.

- **Clusters** – Securing Kubernetes clusters requires securing both the configurable cluster components and the applications that run in the cluster.

- **Containers** – Securing the container layer includes container vulnerability scanning and OS dependency scanning, container image signing and enforcement, and implementing least-privilege access.

- **Code** – Finally, the application code itself must be secured. Security best practices for securing code include requiring TLS for access, limiting communication port ranges, scanning third-party libraries for known security vulnerabilities, and performing static and dynamic code analysis.

3.2.2 DevOps and DevSecOps

In a traditional software-development model, developers write large amounts of code for new features, products, bug fixes, and such, and then pass their work to the Operations team for deployment, usually via an automated ticketing system. The Operations team receives this request in its queue, tests the code, and gets it ready for production – a process that can take days, weeks, or months. Under this traditional model, if Operations runs into any problems during deployment, the team sends a ticket back to the developers to tell them what to fix. Eventually, after this back-and-forth is resolved, the workload gets pushed into production.

This model makes software delivery a lengthy and fragmented process. Developers often see Operations as a roadblock, slowing down their project timelines, while Operations teams feel like the dumping ground for development problems.

DevOps solves these problems by uniting Development and Operations teams throughout the entire software-delivery process, enabling them to discover and remediate issues earlier, automate testing and deployment, and reduce time to market.

To better understand what DevOps is, let's first understand what DevOps is not.

DevOps is not:

- **A combination of the Dev and Ops teams.** There are still two teams; they just operate in a communicative, collaborative way.
- **Its own separate team.** There is no such thing as a "DevOps engineer." Although some companies may appoint a "DevOps team" as a pilot when trying to transition to a DevOps culture, DevOps refers to a culture where developers, testers, and operations personnel cooperate throughout the entire software-delivery lifecycle.
- **A tool or set of tools.** Although there are tools that work well with a DevOps model or help promote DevOps culture, DevOps is ultimately a strategy, not a tool.
- **Automation.** While very important for a DevOps culture, automation alone does not define DevOps.

Now, let's discuss what DevOps is. Instead of developers coding huge feature sets before blindly handing them over to Operations for deployment, in a DevOps model, developers frequently deliver small amounts of code for continuous testing. Instead of communicating issues and requests through a ticketing system, the Development and Operations teams meet regularly, share analytics, and co-own projects from beginning to end.

3.2.2.1 CI/CD pipeline

DevOps is a cycle of continuous integration and continuous delivery (or continuous deployment), otherwise known as the CI/CD pipeline. The CI/CD pipeline integrates Development and Operations teams to improve productivity by automating infrastructure and workflows as well as continuously measuring application performance.

Continuous integration requires developers to integrate code into a repository several times per day for automated testing. Each check-in is verified by an automated build, allowing teams to detect problems early.

Continuous delivery means that the CI pipeline is automated, but the code must go through manual technical checks before it is implemented in production.

Continuous deployment takes continuous delivery one step further. Instead of manual checks, the code passes automated testing and is automatically deployed, giving customers instant access to new features.

3.2.2.2 DevOps and security

One problem in DevOps is that security often ends up falling through the cracks. Developers move quickly, and their workflows are automated. Security is a separate team, and developers don't want to slow down for security checks and requests. As a result, many developers deploy without going through the proper security channels and inevitably make harmful security mistakes.

To solve this, organizations are adopting DevSecOps. DevSecOps takes the concept behind DevOps – the idea that developers and IT teams should work together closely, instead of separately, throughout software delivery – and extends it to include security, integrating automated checks into the full CI/CD pipeline. This takes care of the problem of security seeming like an outside force and allows developers to maintain their speed without compromising data security.

3.2.3 Visibility, governance, and compliance

Ensuring that your cloud resources and SaaS applications are correctly configured and adhere to your organization's security standards from day one is essential to prevent successful attacks. Additionally, making sure these applications, as well as the data they collect and store, are properly protected and compliant is critical to avoid costly fines, brand reputation damage, and loss of customer trust. Meeting security standards and maintaining compliant environments at scale, and across SaaS applications, is a requirement for security teams.

Despite the availability of numerous tools, most organizations struggle to effectively control their data exposure and enforce security policies across ever-changing cloud environments and SaaS applications. Furthermore, ensuring compliance where data is stored across distributed environments puts a significant burden on constrained security teams.

Ensuring governance and compliance across multicloud environments and SaaS applications requires:

- Real-time discovery and classification of resources and data across dynamic SaaS as well as PaaS and IaaS environments

- Configuration governance, ensuring that application and resource configurations match your security best practices as soon as they are deployed and preventing configuration drift
- Access governance using granular policy definitions to govern access to SaaS applications and resources in the public cloud as well as to apply network segmentation
- Compliance auditing, leveraging automation and built-in compliance frameworks, to ensure compliance at any time and generate audit-ready reports on demand
- Seamless user experience that doesn't force additional steps or introduce significant latency in the use of applications as you add new security tools

3.2 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

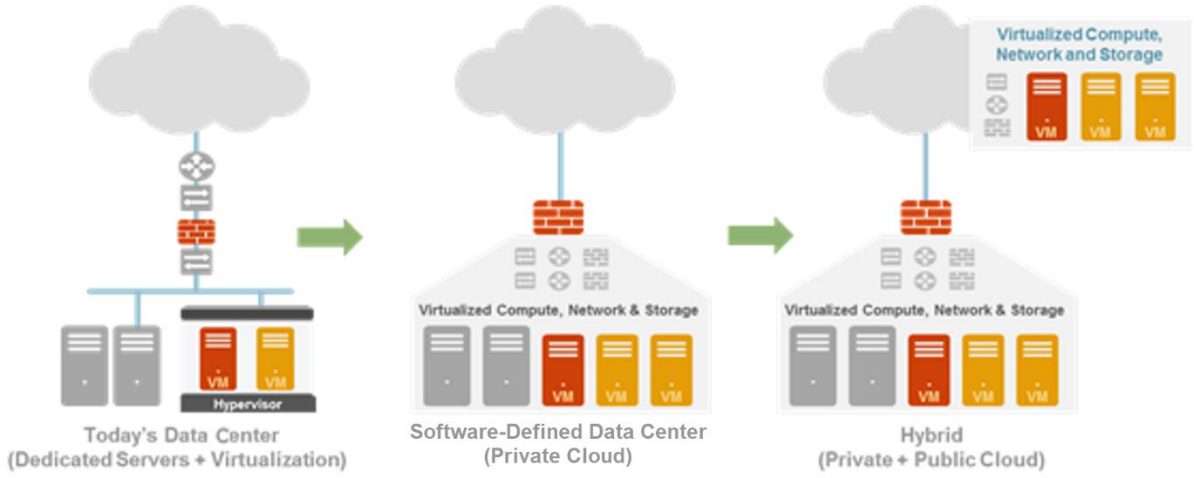
1. **Fill in the Blank.** The "Four C's of cloud-native security" are _____, _____, _____, and _____.
2. **Multiple Choice.** Which of the following processes replaces manual checks with automated code testing and deployment? (Choose one.)
    a) continuous integration
    b) continuous development
    c) continuous delivery
    d) continuous deployment
3. **Fill in the Blank.** _____ requires developers to integrate code into a repository several times per day for automated testing.

3.3 Hybrid Data Center Security

Data centers are rapidly evolving from a traditional, closed environment with static, hardware-based computing resources to an environment in which traditional and cloud computing technologies are mixed (see Figure 3-10).

Figure 3-10

Data centers are evolving to include a mix of hardware and cloud computing technologies.



The benefit of moving toward a cloud computing model – private, public, or hybrid – is that it improves operational efficiencies and lowers capital expenditures:

- **Optimizes existing hardware resources.** Instead of using a "one server, one application" model, you can run multiple virtual applications on a single physical server, which means that organizations can leverage their existing hardware infrastructure by running more applications within the same system, provided that sufficient compute and memory resources exist on the system.

- **Reduces data-center costs.** Reduction of the server hardware "box" count not only reduces the physical infrastructure real estate but also reduces data-center costs for power, cooling, and rack space, among others.

- **Increases operational flexibility.** Through the dynamic nature of provisioning virtual machines (VMs), applications can be delivered more quickly than they can through the traditional method of purchasing them, "racking/stacking," cabling, and so on. This operational flexibility helps improve the agility of the IT organization.

- **Maximizes efficiency of data-center resources.** Because applications can experience asynchronous or bursty demand loads, virtualization provides a more efficient way to address resource contention issues and maximize server use. It also provides a better way to address server maintenance and backup challenges. For example, IT staff can migrate VMs to other virtualized servers or hypervisors while performing hardware or software upgrades.

3.3.1 Traditional data security solution weaknesses

Traditional data-center security solutions exhibit the same weaknesses found when they are deployed at a perimeter gateway on the physical network: They make their initial positive control network-access decisions based on port, using stateful inspection, and then they make a series of sequential, negative control decisions using bolted-on feature sets. This approach has several problems:

- **Limited visibility and control.** The "ports first" focus of traditional data-security solutions limits their ability to see all traffic on all ports, which means that evasive or encrypted applications, and any corresponding threats that may or may not use standard ports, can evade detection. For example, many data-center applications (such as Microsoft Lync, Active Directory, and SharePoint) use a wide range of contiguous ports to function properly. You must therefore open all those ports first, exposing those same ports to other applications or cyberthreats.

- **No concept of unknown traffic.** Unknown traffic is high risk but represents only a relatively small amount of traffic on every network. Unknown traffic can be a custom application, an unidentified commercial off-the-shelf application, or a threat. The common practice of blocking all unknown traffic may cripple your business. Allowing it all is highly risky. You need to be able to systematically manage unknown traffic using native policy-management tools to reduce your organizational security risks.

- **Multiple policies, no policy-reconciliation tools.** Sequential traffic analysis (stateful inspection, application control, intrusion prevention system (IPS), anti-malware, etc.) in traditional data-center security solutions requires a corresponding security policy or profile, often using multiple management tools. The result is that your security policies become convoluted as you build and manage a firewall policy with source, destination, user, port, and action; an application control policy with similar rules; and any other threat prevention rules required. Multiple security policies that mix positive (firewall) and negative (application control, IPS, and anti-malware) control models can cause security holes by missing traffic and/or not identifying the traffic. This situation is made worse when there are no policy-reconciliation tools.

- **Cumbersome security policy update process.** Existing security solutions in the data center do not address the dynamic nature of your cloud environment, because your policies have difficulty contending with the numerous dynamic changes that are common in virtual data centers. In a virtual data center, VM application servers often move from one physical host to another, so your security policies must adapt to changing network conditions.

Many cloud security offerings are merely virtualized versions of port- and protocol-based security appliances with the same inadequacies as their physical counterparts.

3.3.2 East-west traffic protection

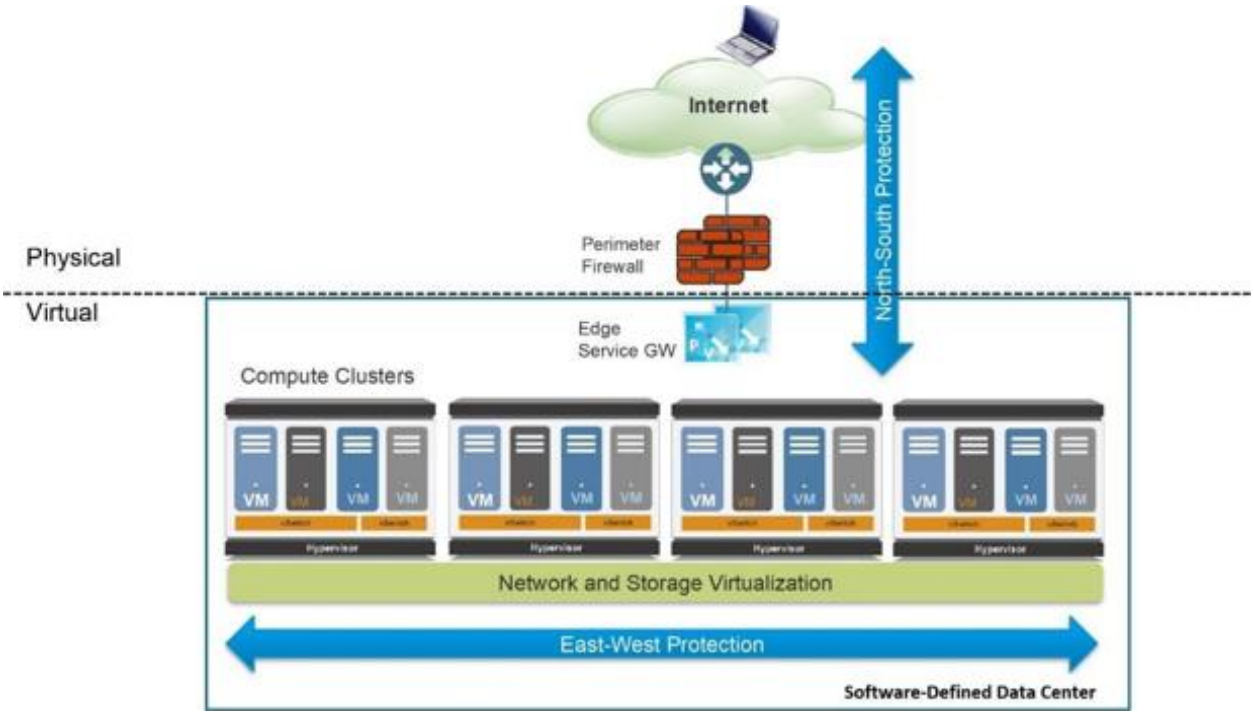
In a virtual data center (private cloud), there are two different types of traffic, each of which is secured in a different manner (see Figure 3-11):

- **North-south** refers to data packets that move in and out of the virtualized environment from the host network or a corresponding traditional data center. North-south traffic is secured by one or more physical-form-factor perimeter-edge firewalls. The edge firewall is usually a high-throughput appliance working in high availability active/passive (or active/active) mode to increase resiliency. It controls all the traffic reaching into the data center and authorizes only allowed and "clean" packets to flow into the virtualized environment.

- **East-west** refers to data packets moving between virtual workloads entirely within the private cloud. East-west traffic is protected by a local, virtualized firewall instantiated on each hypervisor. East-west firewalls are inserted transparently into the application infrastructure and do not necessitate a redesign of the logical topology.

Figure 3-11

Typical virtual data center design architecture



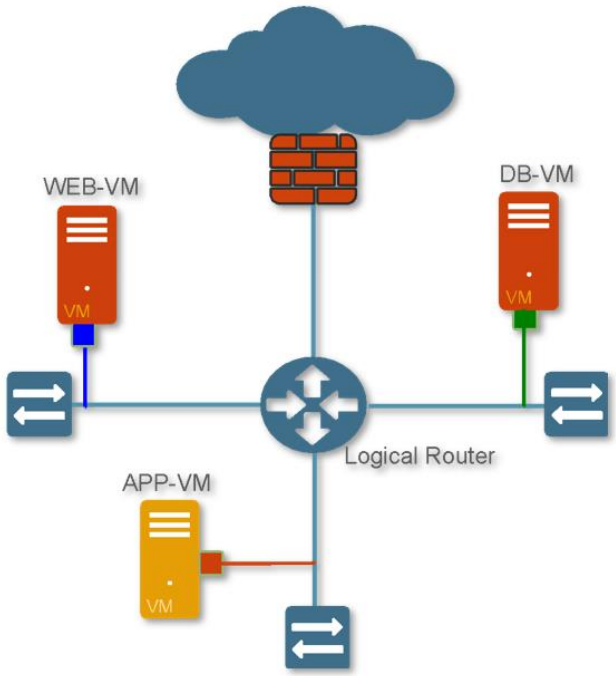
The compute cluster is the building block for hosting the application infrastructure and provides the necessary resources in terms of compute, storage, networking, and security. Compute clusters can be interconnected using OSI model (discussed in Section 2.2.1), Layer 2 (Data Link), or Layer 3 (Network) technologies such as virtual LAN (VLAN), virtual extensible LAN (VXLAN), or Internet Protocol (IP), thus providing a domain extension for workload capacity. Innovations in the virtualization space allow VMs to move freely in this private cloud while preserving compute, storage, networking, and security characteristics and postures.

Organizations usually implement security to protect traffic flowing north-south, but this approach is insufficient for protecting east-west traffic within a private cloud. To improve their security posture, enterprises must protect against threats across the entire network, both north-south and east-west.

One common practice in a private cloud is to isolate VMs into different tiers. Isolation provides clear delineation of application functions and allows a security team to easily implement security policies. Isolation is achieved using logical network attributes (such as a VLAN or a VXLAN) or logical software constructs (such as security groups). Figure 3-12 displays a simple three-tier application that is composed of a WEB-VM as the front end, an APP-VM as the application, and a DB-VM providing database services.

Figure 3-12

Three-tier application hosted in a virtual data center



An attacker has multiple options to steal data from the DB-VM. The first option is to initiate a SQL injection attack by sending HTTP requests containing normalized SQL commands that target an application vulnerability. The second option is to compromise the WEB-VM (using vulnerability exploits) and then move laterally to the APP-VM, initiating a brute-force attack to retrieve the SQL admin password.

After the DB-VM is compromised, the attacker can hide sensitive data extraction by using techniques such as DNS tunneling or by moving data across the network with NetBIOS and then off the network via FTP. In fact, attackers using applications commonly found on nearly every network have virtually unlimited options for stealing critical data in this environment. Infiltration into the environment and exfiltration of critical data can be completely transparent and undetected because the data is carried over legitimate protocols (such as HTTP and DNS) that are used for normal business activities.

Virtual data center security best practices dictate a combination of north-south and east-west protection. East-west protection provides the following benefits:

- Authorizes only allowed applications to flow inside the data center, between VMs
- Reduces lateral threat movement when a front-end workload has been compromised (the attacker breaches the front-end server by using a misconfigured application or unpatched exploit)
- Stops known and unknown threats that are sourced internally within the data center
- Protects against data theft by leveraging data and file-filtering capability and blocking anti-spyware communications to the external world

An added benefit of using virtual firewalls for east-west protection is the unprecedented traffic and threat visibility that the virtualized security device can now provide. After traffic logs and threat logs are turned on, VM-to-VM communications and malicious attacks become visible. This virtual-data-center awareness allows security teams to optimize policies and enforce cyberthreat protection (for example, IPS, anti-malware, file blocking, data filtering, and DoS protection) where needed.

3.3.3 Security in hybrid data centers

The following approach to security in the evolving data center – from traditional three-tier architectures to virtual data centers and to the cloud – aligns with practical realities, such as the need to leverage existing best practices and technology investments, and the likelihood that most organizations will transform their data centers incrementally.

This approach consists of four phases:

- **Consolidating servers within trust levels.** Organizations often consolidate servers within the same trust level into a single virtual computing environment: either one physical host or a cluster of physical hosts. Intrahost communications are generally minimal and inconsequential. As a matter of routine, most traffic is directed "off box" to users and systems residing at different trust levels. When intrahost communications do happen, the absence of protective safeguards between these virtualized systems is also consistent with the organization's security posture for nonvirtualized systems. Live migration features are typically used to enable transfer of VMs only to hosts supporting workloads within the same subnet. Security solutions should incorporate a robust virtual-systems capability in which a single instance of the associated countermeasures can be partitioned into multiple logical instances, each with its own policy, management, and event domains. This virtual-systems capability enables a single physical device to be used to simultaneously meet the unique requirements of multiple VMs or groups of VMs. Controlling and protecting interhost traffic with physical network-security appliances that are properly positioned and configured is the primary security focus.

- **Consolidating servers across trust levels.** Workloads with different trust levels often coexist on the same physical host or cluster of physical hosts. Intrahost communications are limited, and live migration features are used to enable transfer of VMs only to hosts that are on the same subnet and that are configured identically with regard to routing of VM-to-VM traffic. Intrahost communication paths are intentionally not configured between VMs with different trust levels. Instead, all traffic is forced off box through a default gateway – such as a physical network-security appliance – before it is allowed to proceed to the destination VM. Typically, this off-box routing can be accomplished by configuring separate virtual switches with separate physical network interface cards (NICs) for the VMs at each distinct trust level. As a best practice for virtualization, you should minimize the combination of workloads with different trust levels on the same server. Live migrations of VMs also should be restricted to servers supporting workloads within the same trust levels and within the same subnet. Over time, and in particular as workloads move to the cloud, maintenance of segmentation based on trust levels becomes more challenging.

- **Selective network security virtualization.** Intrahost communications and live migrations are architected at this phase. All intrahost communication paths are strictly controlled to ensure that traffic between VMs at different trust levels is intermediated either by an on-box, virtual security appliance or by an off-box, physical security appliance. Long-

distance live migrations (for example, between data centers) are enabled by a combination of native live-migration features with external solutions that address associated networking and performance challenges. The intense processing requirements of solutions such as next-generation firewall virtual appliances will ensure that purpose-built physical appliances continue to play a key role in the virtual data center. However, virtual instances are ideally suited for scenarios where countermeasures need to migrate along with the workloads they control and protect.

- **Dynamic computing fabric.** Conventional, static computing environments are transformed into dynamic fabrics (private or hybrid clouds) where underlying resources such as network devices, storage, and servers can be fluidly engaged in whatever combination best meets the needs of the organization at any given point in time. Intrahost communication and live migrations are unrestricted. This phase requires networking and security solutions that not only can be virtualized but are also virtualization-aware and can dynamically adjust as necessary to address communication and protection requirements, respectively. Classification, inspection, and control mechanisms in virtualization-aware security solutions must not be dependent on physical and fixed Network-layer attributes. In general, higher-layer attributes such as application, user, and content identification are the basis not only for how countermeasures deliver protection but also for how they dynamically adjust to account for whatever combination of workloads and computing resources exist in their sphere of influence. Associated security-management applications also need to be capable of orchestrating the activities of physical and virtual instances of countermeasures first with each other and then with other infrastructure components. This capability is necessary to ensure that adequate protection is optimally delivered in situations where workloads are frequently migrating across data-center hosts.

3.3 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Short Answer.** List some of the principles of cloud computing that are contrary to network security best practices.
2. **Multiple Choice.** Intra-VM traffic is also known as which type of traffic? (Choose one.)
    a) north-south
    b) unknown
    c) east-west
    d) untrusted
3. **Multiple Choice.** What is the first phase of implementing security in virtualized data centers? (Choose one.)
    a) consolidating servers across trust levels
    b) consolidating servers within trust levels
    c) selectively virtualizing network-security functions
    d) implementing a dynamic computing fabric

3.4 Secure the Cloud (Prisma)

Application-development methodologies are moving away from the traditional "waterfall" model toward more agile continuous integration/continuous delivery (CI/CD) processes with end-to-end automation. This new approach brings a multitude of benefits, such as shorter time to market and faster delivery, but it also introduces security challenges because traditional security methodologies weren't designed to address these modern application workflows. As developer teams embrace cloud-native technologies, security teams find themselves scrambling to keep up. Limited prevention controls, poor visibility, and tools that lack automation yield incomplete security analytics; all of these things increase the risk of compromise and the likelihood of successful breaches in cloud environments. Meanwhile, the demand for an entirely new approach to security emerges. Enter cloud-native security platforms (CNSPs).

The term “cloud native” refers to an approach to building and running applications that takes full advantage of a cloud computing delivery model instead of an on-premises data center. This approach takes the best of what cloud has to offer – scalability, deployability, manageability, and limitless on-demand compute power – and applies these principles to software development, combined with CI/CD automation, to radically increase productivity, business agility, and cost savings.

Cloud-native architectures consist of cloud services, such as containers, serverless security, platform as a service (PaaS), and microservices. These services are loosely coupled, meaning they are not hardwired to any infrastructure components, allowing developers to make changes frequently without affecting other pieces of the application or other team members’ projects – all across technology boundaries, such as public, private, and multicloud deployments.

In short, “cloud native” refers to a methodology of software development that is essentially designed for cloud delivery and exemplifies all the benefits of the cloud by nature.

As more organizations have embraced DevOps and developer teams have begun to update their application-development pipelines, security teams quickly realized that their tools were ill-suited for the developer-driven, API-centric, infrastructure-agnostic patterns of cloud-native security. As a result, cloud-native security point products began to hit the market. These products were each engineered to address one part of the problem or one segment of the software stack, but on their own they could not collect enough information to accurately understand or report on the risks across cloud-native environments. This situation forced security teams to juggle multiple tools and vendors, which increased cost, complexity, and risk in addition to creating blind spots where the tools overlapped but didn’t integrate.

Solving this problem requires a unified platform approach that can envelop the entire CI/CD lifecycle and integrate with the DevOps workflow. Just as cloud-native approaches have fundamentally changed the how cloud is used, CNSPs are fundamentally restructuring how the cloud is secured.

CNSPs share context about infrastructure, PaaS, users, development platforms, data, and application workloads across platform components to enhance security. They also:

- Provide unified visibility for SecOps and DevOps teams
- Deliver an integrated set of capabilities to respond to threats and protect cloud-native applications
- Automate the remediation of vulnerabilities and misconfigurations consistently across the entire build-deploy-run lifecycle

In the past, organizations that wanted to embrace new compute options were stifled by the need to buy more security products to support those options. Stitching together disparate solutions in an attempt to enforce consistent policies across technology boundaries became more of a problem than a solution. CNSPs, however, provide coverage across the continuum of compute options, multicloud, and the application-development lifecycle. This coverage allows organizations to choose the right compute options for any given workload, granting them freedom without worry over how to integrate solutions for security. CNSPs epitomize the benefits of a cloud-native strategy, enabling agility, flexibility, and digital transformation.

The Palo Alto Networks CNSP includes the following solutions to secure the cloud: Prisma Cloud, Prisma Access, and Prisma SaaS.

3.4.1 Cloud application security (Prisma Cloud)

Prisma Cloud is the most comprehensive cloud-native security platform, designed to protect all aspects of cloud usage with the industry's leading technology. Prisma Cloud provides broad security and compliance coverage for the entire cloud-native technology stack, as well as applications and data throughout the entire application lifecycle, across multicloud and hybrid cloud environments. Prisma Cloud takes an integrated approach that enables SecOps and DevOps teams to accelerate cloud-native application deployment by implementing security early in the development cycle.

Prisma Cloud rests on four pillars:

- **Visibility, governance, and compliance.** Gain deep visibility into the security posture of multicloud environments. Keep track of everything that gets deployed with an automated asset inventory, and maintain compliance with out-of-the-box governance policies that enforce good behavior across your environments.
- **Compute security.** Secure hosts, containers, and serverless workloads throughout the application lifecycle. Detect and prevent risks by integrating vulnerability intelligence into your *integrated development environment* (IDE), *software configuration management* (SCM), and CI/CD workflows. Enforce machine-learning-based runtime protection to protect applications and workloads in real time.
- **Network protection.** Continuously monitor network activity for anomalous behavior, enforce microservice-aware microsegmentation, and implement industry-leading firewall protection. Protect the network perimeter as well as the connectivity between containers and hosts.

- **Identity security.** Monitor and leverage *user and entity behavior analytics (UEBA)* across your environments to detect and block malicious actions. Gain visibility into, and enforce, governance policies on user activities, and manage the permissions of both users and workloads.

Key Terms

An *integrated development environment* (IDE) is a software application that provides comprehensive tools – such as a source-code editor, build automation tools, and a debugger – for application developers.

Software configuration management (SCM) is the task of tracking and controlling changes in software.

User and entity behavior analytics (UEBA) is a type of cybersecurity solution or feature that discovers threats by identifying activity that deviates from a baseline.

3.4.1.1 Cloud governance and compliance

Ensuring that your cloud resources and SaaS applications are correctly configured and adhere to your organization's security standards from day one is essential to prevent successful attacks. Additionally, making sure that these applications, as well as the data they collect and store, are properly protected and compliant is critical to avoid costly fines, a tarnished image, and loss of customer trust. Meeting security standards and maintaining compliant environments at scale, and across SaaS applications, is the new expectation for security teams.

Despite the availability of numerous tools, most organizations struggle to effectively control their data exposure and enforce security policies across ever-changing cloud environments and SaaS applications. Furthermore, ensuring compliance where data is stored across distributed environments puts a significant burden on your already constrained security teams.

Ensuring governance and compliance across multicloud environments and SaaS applications requires:

- **Real-time discovery and classification** of resources and data across dynamic SaaS, PaaS, and IaaS environments
- **Configuration governance** ensuring that application and resource configurations match your security best practices as soon as they are deployed, and preventing configuration drift

- **Access governance** using granular policy definitions to govern access to SaaS applications and resources in the public cloud as well as to apply network segmentation
- **Compliance auditing** leveraging automation and built-in compliance frameworks, to ensure compliance at any time and generate audit-ready reports on demand
- **Seamless user experience** that doesn't force additional steps or introduce significant latency in the use of applications as you add new security tools

3.4.1.2 Compute security

The cloud-native landscape is constantly evolving with new technologies and levels of abstraction. Hosts, containers, and serverless workloads provide unique benefits and have different security requirements. Prisma Cloud provides best-in-class solutions for securing any type of cloud-native workload, throughout the development lifecycle.

Prisma Cloud provides cloud-native compute security from build to run, including:

- **Vulnerability management.** Detect and prevent vulnerabilities and misconfigurations throughout the entire development process. Prioritize vulnerabilities based on your unique environment and prevent vulnerable code from ever reaching production.
- **Runtime security.** Prevent threats and anomalies across your hosts, containers, serverless functions, and orchestrators. Build automated, machine-learning-driven models that define known good behaviors across process, network, file system, and system call sensors. Models are correlated to image IDs, so every time you build your app, you get a model uniquely calculated and tailored for that specific build.
- **Application security.** Protect applications and APIs through a powerful combination of web traffic inspection and *runtime application self-protection* (RASP). Embrace an "explicit allow" model where only the specific activities and capabilities required by your application are allowed – and everything else is treated as anomalous and is therefore prevented.
- **DevSecOps enabled.** Integrate security into your IDE, SCM, and CI workflows to detect and prevent issues as early as possible. Powerful plugins allow developers to inspect images, IaC templates, and functions as well as see vulnerability status every time they run a build. Security teams can prevent compromised assets from ever progressing down the pipeline.

Key Terms

Runtime application self-protection (RASP) detects attacks against an application in real time. RASP continuously monitors an app's behavior and the context of behavior to immediately identify and prevent malicious activity.

3.4.1.3 Network protection

Network protection must be adapted for cloud-native environments while still enforcing consistent policies across hybrid environments. Prisma Cloud detects and prevents network anomalies by enforcing container-level microsegmentation, inspecting traffic flow logs, and leveraging advanced Layer 7 threat protection.

Prisma Cloud network protection capabilities include:

- **Network visibility and anomaly detection.** Ingest network traffic flow logs from multiple sources and gain deep visibility into network behavior to detect and prevent anomalies.
- **Identity-based microsegmentation.** Enforce cloud-native microsegmentation at the container and host levels with Layer 4 and Layer 7 distributed firewalls. Segment cloud networks and deploy policies based on logical workload and application identities, rather than dynamic IP addresses.
- **Cloud-native firewalling.** Automatically model traffic flows between microservices and dynamically create filters that allow valid connections and drop suspicious ones. Protect networks with Layer 4 and Layer 7 security capabilities, such as DNS security and URL filtering.

3.4.1.4 Identity security

Managing a large number of privileged users with access to an ever-expanding set of sensitive resources can be challenging. On top of that, cloud resources themselves have permission sets that need to be managed. Prisma Cloud helps you leverage the identity of cloud resources to enforce security policies and ensure secure user behavior across your cloud environments.

Key capabilities include:

- **Identity and access management (IAM) security.** Secure and manage the relationships between users and cloud resources. Enforce governance policies to ensure that users and resources behave only as intended and do not introduce risk to the environment.

- **Access management.** Ensure least-privileged access to cloud resources and infrastructure, and decouple user permissions from workload permissions.
- **Machine identity.** Decouple workload identity from IP addresses. Leverage tags and metadata to assign a logical identity to applications and workloads, and then use it to enforce ID-based microsegmentation and security policies that adapt to your dynamic environments.
- **UEBA.** Continuously analyze the behavior of users and resources in your cloud to detect and prevent anomalous behavior, such as an admin logging in from an unknown location or a container accessing a file it should not be able to access.

3.4.2 Secure Access Service Edge (Prisma Access)

With increasing numbers of mobile users, branch offices, data, and services located outside the protections of traditional network security appliances, organizations are struggling to keep pace and ensure the security, privacy, and integrity of their networks and their customers' data.

Today, many of the technologies on the market are built upon architectures that were not designed to handle all types of traffic and security threats. This forces organizations to adopt multiple point products to handle different requirements, such as secure web gateways, firewalls, secure VPN remote access, and SD-WAN. For every product, there is an architecture to deploy, a set of policies to configure, and an interface to manage, each with its own set of logs. This situation creates an administrative burden that introduces cost, complexity, and gaps in security posture.

To address these challenges, Secure Access Service Edge (SASE) has emerged. SASE (pronounced "sassy") is designed to help organizations embrace cloud and mobility by providing networking and network security services from a common cloud-delivered architecture. A SASE solution must provide consistent security services and access to all types of cloud applications (public cloud, private cloud, and SaaS) delivered through a common framework. By removing multiple point products and adopting a single cloud-delivered SASE solution, organizations can reduce complexity while saving significant technical, human, and financial resources.

A SASE solution converges networking and security services into one unified, cloud-delivered solution (see Figure 3-13) that includes the following:

- Networking:
    - Software-defined wide-area networks (SD-WANs)

    - Virtual private networks (VPNs)
    - Zero Trust network access (ZTNA)
    - Quality of service (QoS)
- Security:
    - Firewall as a service (FWaaS)
    - Domain Name System (DNS) security
    - Threat prevention
    - *Secure web gateway* (SWG)
    - Data loss prevention (DLP)
    - *Cloud access security broker* (CASB)

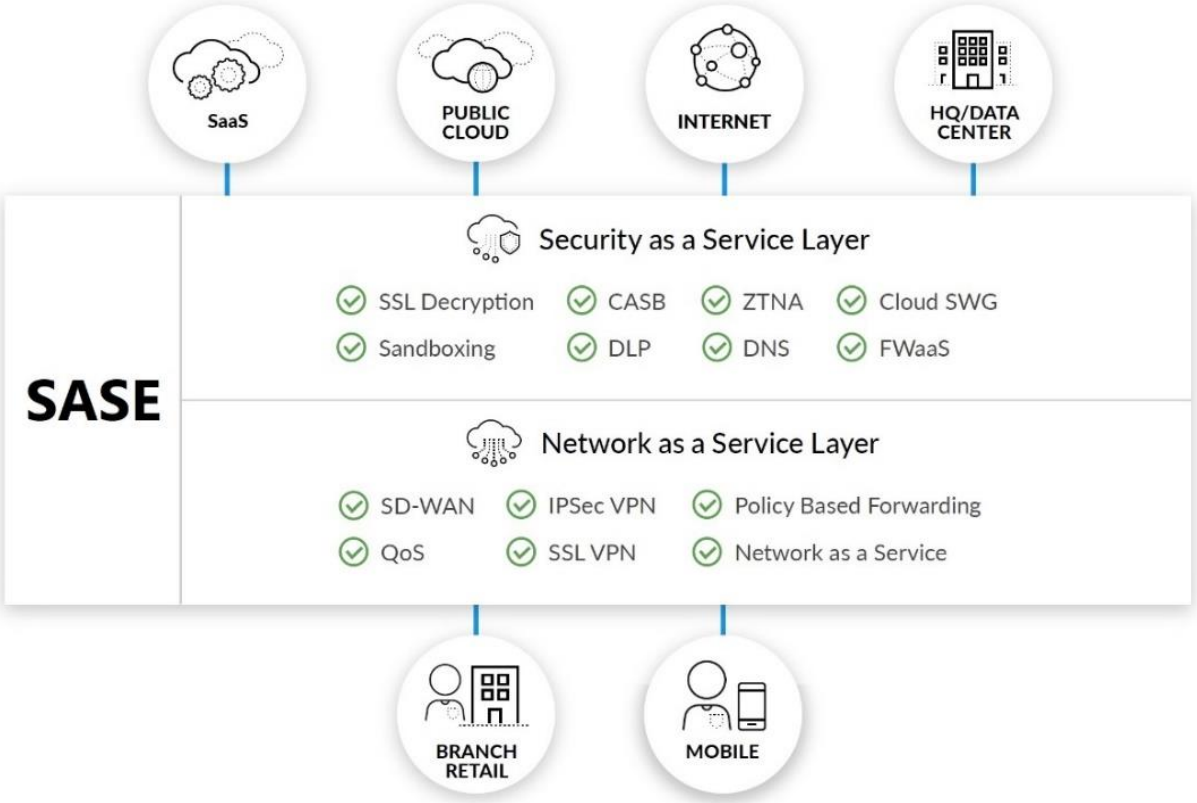
Key Terms

A *secure web gateway* (SWG) is a security platform or service that is designed to maintain visibility in web traffic. Additional functionality may include web content filtering.

A *cloud access security broker* (CASB) is software that monitors activity and enforces security policies on traffic between an organization's users and cloud-based applications and services.

Figure 3-13

SASE delivers advanced network and security capabilities in a converged, cloud-delivered solution.



Prisma Access delivers globally distributed networking and security to all your users and applications. Whether at branch offices or on the go, your users connect to Prisma Access to safely access cloud and data-center applications as well as the internet.

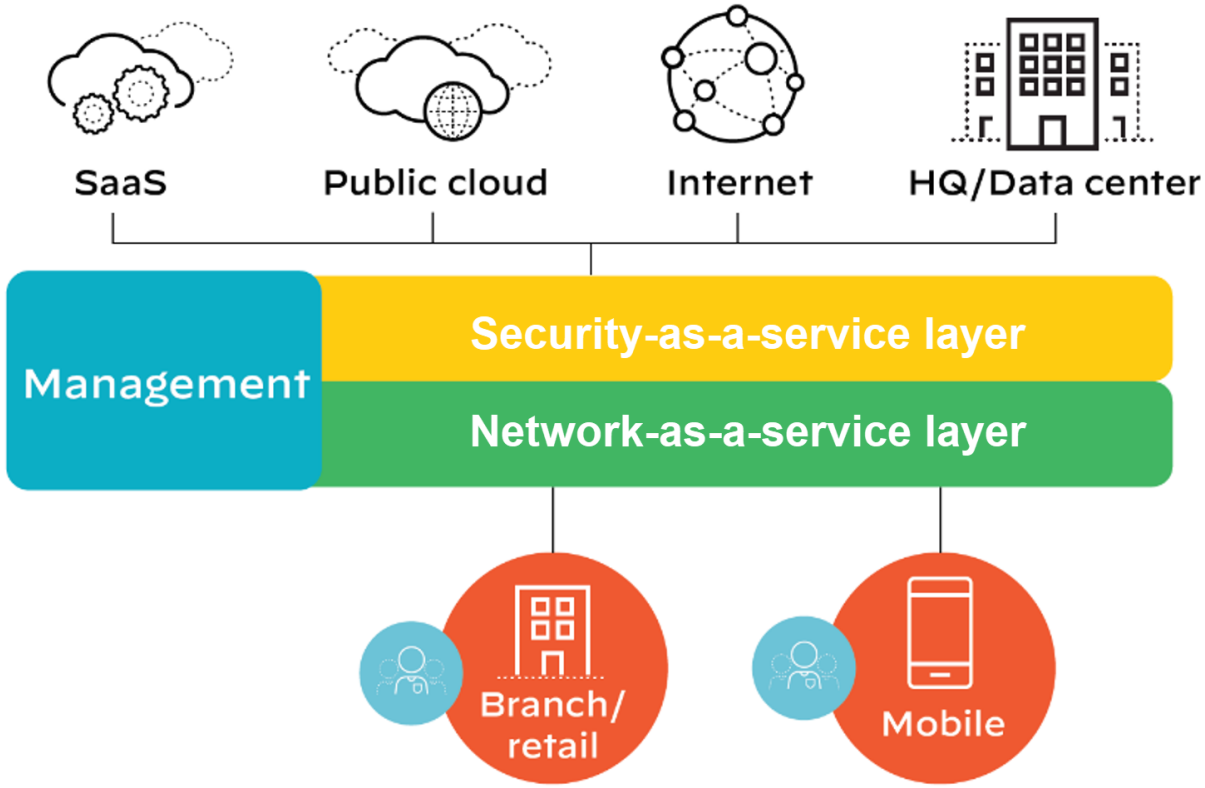
Prisma Access consistently protects all traffic, on all ports and from all applications, enabling your organization to:

- **Prevent successful cyberattacks** with proven security philosophies and threat intelligence for deep visibility and precise control that extends across your organization
- **Fully inspect all application traffic** bidirectionally – including SSL/TLS-encrypted traffic – on all ports, whether communicating with the internet, with the cloud, or between branches
- **Benefit from comprehensive threat intelligence** powered by automated threat data from Palo Alto Networks and hundreds of third-party feeds

The Prisma Access SASE architecture consists of a network-as-a-service layer, a security-as-a-service layer, and a common management platform to secure branch/retail and mobile users across SaaS, public cloud, internet, and headquarters/data-center environments (see Figure 3-14).

Figure 3-14

The Prisma Access architecture



3.4.2.1 Network-as-a-service layer

The network-as-a-service layer in Prisma Access delivers key SASE capabilities, including:

- Software-defined wide-area network (SD-WAN)
- Virtual private network (VPN)
- Zero Trust network access (ZTNA)
- Quality of service (QoS)

SD-WAN

Companies are embracing software-defined wide-area networking (SD-WAN) to connect branch offices to the corporate network and provide local internet breakout as an alternative to costly multiprotocol label switching (MPLS) connections. The challenge with SD-WAN, however, is how to combine security with the SD-WAN fabric, which leads to the need for multiple overlays.

In a SASE solution, SD-WAN edge devices can be connected to a cloud-based infrastructure, rather than to physical SD-WAN hubs located in data-center or collocation facilities. This approach enables the interconnectivity between branch offices without the complexity of deploying and managing physical SD-WAN hubs.

SD-WAN should be something you are already considering or have already adopted into your organization's network infrastructure as a way to securely connect and control access to branch offices and remote employees. SASE creates a unified framework for SD-WAN services and other solutions to connect to, providing a single point of view and simplified management solution to protect your network.

Prisma Access connects branch offices over a standard IPSec VPN tunnel using common IPSec-compatible devices, such as your existing branch router, SD-WAN edge device, or a third-party firewall. It uses Border Gateway Protocol (BGP) or static routes for routing from the branch and equal-cost multi-path (ECMP) routing for faster performance and better redundancy across multiple links.

Virtual private network

Organizations rely on virtual private networks (VPNs) to provide a secure encrypted connection for mobile users and branch offices to access corporate data, applications, and internet access. There are many types of VPN services – from IPSec VPN to SSL VPN, clientless VPN, and remote-access VPN – all of which require a connection to a VPN gateway. VPNs are not optimized for access to the cloud, resulting in no security or access control when users disconnect to reach cloud apps or services.

A SASE solution encompasses VPN services and enhances the capabilities to operate in a cloud-based infrastructure in order to securely route traffic to the public cloud, SaaS, internet, or private-cloud apps. In an IPSec VPN, for example, you can create a site-to-site connection to a cloud-based infrastructure from any IPSec-compatible device located at a branch or retail location via a branch router, wireless access point, SD-WAN edge device, or firewall. Mobile users employ an always-on IPSec or SSL VPN connection between their endpoint or mobile device and a SASE solution, which ensures consistent traffic encryption and threat prevention.

No matter which type of VPN service you use in your organization, a SASE solution provides a unified cloud infrastructure to connect to, instead of backhauling to a VPN gateway at corporate headquarters. This solution dramatically simplifies the management and policy control needed to enforce least-privileged access rules.

Prisma Access (formerly GlobalProtect cloud service) provides cloud-delivered security infrastructure that makes it possible for your organization to connect users to a nearby cloud gateway, enable secure access to all applications, and maintain full visibility and inspection of traffic across all ports and protocols.

For managed mobile devices:

- Users with managed devices have the GlobalProtect app installed on their laptop, mobile phone, or tablet. The GlobalProtect app connects to Prisma Access automatically whenever internet access is available, without requiring any user interaction.
- Users can access all of their applications, whether in the cloud or the data center. The connectivity layer connects applications in different locations, making it possible to establish secure access (based on App-ID and User-ID policies) to public cloud, SaaS, and data-center applications.
- Prisma Access delivers protection through the security service layer, such as protections against known and unknown malware, exploits, C2 traffic, and credential-based attacks.

For unmanaged/BYOD devices:

- Your organization can deploy Prisma Access in conjunction with mobile device management (MDM) integration to support bring-your-own-device (BYOD) policies. The integration enables capabilities such as per-app VPN.
- Users with unmanaged devices, such as contractors and employees with BYOD devices, can access applications without an app installed by using Prisma Access with Clientless VPN.
- Clientless VPN also enables secure access to SaaS applications from unmanaged devices with inline protections by using Security Assertion Markup Language (SAML) proxy integration. This functionality works in conjunction with Prisma SaaS.

Zero Trust network access

Zero Trust network access (ZTNA) is a key part of the Zero Trust philosophy of “never trust, always verify,” developed by Forrester to identify the need to protect data. ZTNA requires users

who want to connect to the cloud to authenticate through a gateway before gaining access to the applications they need. This requirement provides an IT admin the ability to identify users and create policies to restrict access, minimize data loss, and quickly mitigate any issues or threats that may arise.

Many ZTNA products are based on software-defined perimeter (SDP) architectures, which do not provide content inspection, thus creating a discrepancy in the types of protection available for each application. In terms of consistent protection, the organization must build additional controls on top of the ZTNA model and establish inspection for all traffic across all applications.

SASE builds on the ZTNA key principles and applies them across all the other services within a SASE solution. Identifying users, devices, and applications, no matter where they are connecting from, simplifies policy creation and management. SASE removes the complexity of connecting to a gateway by incorporating the networking services into a single unified cloud infrastructure.

A SASE solution should incorporate ZTNA concepts for protecting applications as well as apply other security services for the consistent enforcement of DLP and threat prevention policies. Access controls, in and of themselves, are useful for establishing who a person is, but other security controls are also necessary to make sure that their behaviors and actions are not harmful to the organization. And it is necessary to apply the same controls across access to all applications.

Quality of service

As organizations transition from MPLS to SD-WAN using broadband services, they are finding that the service quality varies. Quality of service (QoS) establishes bandwidth allocation to particular apps and services. Businesses rely on QoS to ensure that their critical apps and services perform adequately (for example, medical equipment or credit-card processing services). If these systems were to get bogged down due to lack of bandwidth, business operations and sales would be severely impacted. QoS prioritizes business-critical apps, based on a ranking system, so that you can choose which apps and services take precedence over others.

QoS is an important step when you begin migrating from MPLS. A SASE solution incorporates QoS services in the cloud, allowing you to easily mark sensitive applications (such as VoIP) as higher priority than general internet browsing and entertainment apps.

QoS is immensely important for businesses of any size. Managing the QoS traffic and allocation doesn’t need to be difficult. SASE enables you to dynamically shape traffic based on the policies that prioritize critical application requirements. Make sure that your SASE solution contains QoS capabilities.

3.4.2.2 Security-as-a-service layer

The security-as-a-service layer in Prisma Access delivers key SASE capabilities, including:

- DNS security
- Firewall as a service (FWaaS)
- Threat prevention
- Secure web gateway (SWG)
- Data loss prevention (DLP)
- Cloud access security broker (CASB)

DNS security

Every organization uses DNS to translate a domain name into an IP address. DNS is an open service, and by default it does not have a way to detect DNS-based threats. As a result, malicious activity within DNS can be used to propagate an attack.

DNS security protects your users by predicting and blocking malicious domains while neutralizing threats. A SASE solution embraces DNS security features by providing consistent security across the network and users, no matter their location.

Your SASE solution should contain DNS protections, delivered within the cloud environment as part of the network access. DNS security should be built in, rather than bolted on, to the solution your branch offices and mobile users use to connect to the internet. The DNS security provided in your SASE solution should leverage a combination of predictive analytics, machine learning, and automation to combat threats in DNS traffic.

Prisma Access delivers the Palo Alto Networks DNS Security service (discussed in Section 2.6.2.1), which provides a combination of predictive analytics, machine learning, and automation to combat threats in DNS traffic. Organizations can block known malicious domains, predict new malicious domains, and stop DNS tunneling.

Firewall as a service

Firewall as a service (FWaaS) is a deployment method for delivering a firewall as a cloud-based service. FWaaS has the same features as a next-generation firewall, but it is implemented in the cloud. By moving the firewall to the cloud, organizations can benefit from cost savings by eliminating the need to install or maintain security hardware at branch and retail locations.

A SASE solution incorporates FWaaS into its unified platform. By encompassing the FWaaS service model within a SASE framework, organizations can easily manage their deployments from a single platform.

A SASE solution should enable FWaaS capabilities in order to provide the protection of a next-generation firewall by implementing Network Security policy in the cloud. It is important to ensure that your SASE solution does not provide only basic port blocking or minimal firewall protections. You need the same features that a next-generation firewall embodies as well as the features that cloud-based security offers, such as threat prevention services and DNS security.

Prisma Access provides FWaaS, which protects branch offices from threats while also providing the security services expected from a next-generation firewall. The full spectrum of FWaaS includes threat prevention, URL filtering, sandboxing, and more.

Threat prevention

In today's world of small- and large-scale breaches, where ransomware attacks occur on a daily basis, threat prevention is key to protecting your organization's data and employees. A variety of threat prevention tools are available, from anti-malware and intrusion prevention to SSL decryption and file blocking, providing organizations with ways to block threats. However, these point products require separate solutions, making management and integration difficult.

Within a SASE solution, all these point products and services are now integrated into a single cloud platform. This integration provides simplified management and oversight of all threats and vulnerabilities across your network and cloud environments.

Stopping exploits and malware by using the latest threat intelligence is crucial to protecting your employees and data. Your SASE solution should incorporate threat prevention tools into its framework so that you can react swiftly to remediate threats. Be sure to check the quality of threat intelligence that is being provided by the vendor. The vendor should be gathering and sharing data from various sources, including customers, other vendors, and other related thought leaders, to provide continuous protection from unknown threats.

Using Prisma Access for threat prevention combines the proven technologies in the Palo Alto Networks platform, together with global sources of threat intelligence and automation, to stop previously known or unknown attacks.

Secure web gateway

Organizations rely on a secure web gateway (SWG) to protect employees and devices from accessing malicious websites. An SWG can be used to block inappropriate content (such as

pornography and gambling) or websites that businesses simply don't want users accessing while at work, such as streaming services like Netflix. Additionally, an SWG can be used to enforce an acceptable use policy (AUP) before internet access is granted.

An SWG is just one of the many security services that a SASE solution must provide. As organizations grow and add ever greater numbers of remote users, coverage and protection become more difficult. A SASE solution moves the SWG into the cloud, providing protection in the cloud through a unified platform for complete visibility and control over the entire network.

A SASE solution includes the same security services in a SWG, allowing organizations to control access to the web and enforce security policies that protect users from hostile websites. It is important to remember that an SWG is just one service of the SASE solution. Other security services – such as FWaaS, DNS Security, Threat Prevention, DLP, and CASB – should also be included.

Prisma Access for SWG functionality is designed to maintain visibility into all types of traffic while stopping evasions that can mask threats. The Palo Alto Networks web-filtering capabilities also drive the company's credential-theft-prevention technology, which can stop corporate credentials from being sent to previously unknown sites.

Data loss prevention

Data loss prevention (DLP) tools protect sensitive data and ensure that it is not lost, stolen, or misused. DLP is a composite solution that monitors data within the environments where it is deployed (such as networks, endpoints, and clouds) and through their egress points. It also alerts key stakeholders when policies are violated. Due to compliance requirements – such as the Health Insurance Portability and Accountability Act (HIPAA), Payment Card Industry Data Security Standard (PCI DSS), General Data Protection Regulation (GDPR), and others – DLP is a crucial solution needed for data security and compliance. Legacy DLPs rely on old core technology initially designed for on-premises perimeters and subsequently extended and adapted to cloud applications. Loaded with features, disjointed policies, configurations, and workarounds, DLPs have become very complex, difficult to deploy at scale, and too expensive. Digital transformation and new data-usage models demand a fresh approach to data protection.

Through the SASE approach, DLP becomes one cloud-delivered solution centered around the data itself, everywhere. The same policies are consistently applied to sensitive data, whether at rest, in motion, or in use, and regardless of its location. In the SASE architecture, DLP is not a standalone solution anymore but is embedded in the organization's existing control points, thus eliminating the need to deploy and maintain multiple tools. With SASE, organizations can finally

enable a comprehensive data-protection solution that relies on a scalable and simple architecture and allows effective machine learning by leveraging access to global traffic.

DLP is a necessary tool to protect sensitive data and ensure compliance throughout the organization. To this end, the SASE solution must include this core capability. With SASE, DLP is an embedded, cloud-delivered service used to accurately and consistently identify, monitor, and protect sensitive data everywhere across networks, clouds, and users.

Prisma Access combines integration with DLP controls that are API-driven (through Prisma SaaS) as well as inline (through Prisma Access). These DLP policies allow organizations to categorize data and establish policies that prevent data loss.

Cloud access security broker

Many organizations depend on cloud access security brokers (CASBs) to provide visibility into SaaS application usage, understand where their sensitive data resides, enforce company policies for user access, and protect their data from hackers. CASBs are cloud-based security-policy enforcement points that provide a gateway for your SaaS provider and your employees.

A CASB should be another security feature that is part of your SASE solution, creating a single platform for stakeholders to manage security controls. A SASE solution helps you understand which SaaS apps are being used and where data is going, no matter where users are located.

Your SASE solution should incorporate both inline and API-based SaaS controls for governance, access controls, and data protection. Also called a multimode CASB, the combination of inline and API-based CASB capabilities provides superior visibility, management, security, and zero-day protection against emerging threats.

Prisma Access and Prisma SaaS implement security controls that combine inline security API security and contextual controls, acting as a CASB to determine access to sensitive information. These controls are implemented in an integrated manner and applied throughout all cloud application policies.

3.4.3 Prisma SaaS

To safely enable SaaS usage in your organization, start by clearly defining the SaaS applications that should be used and which behaviors within those applications are allowed. This step requires a clear definition of which applications are:

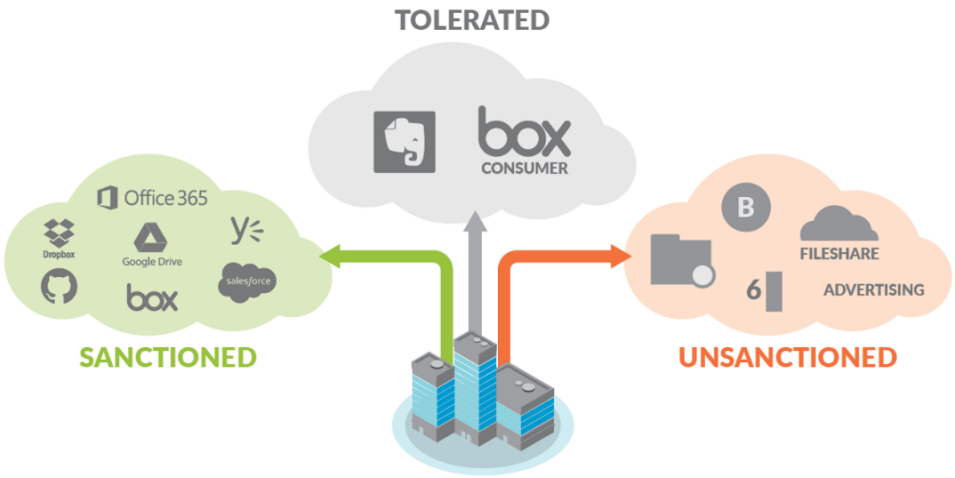
- **Sanctioned** (allowed and provided by IT)

- **Tolerated** (allowed because of a legitimate business need, with restrictions, but not provided by IT)
- **Unsanctioned** (not allowed); prevent their usage with granular policies

Sanctioned SaaS applications provide business benefits, and they are fast to deploy, require minimal cost, and are infinitely scalable. Tolerated SaaS applications fulfill a legitimate business need, but certain usage restrictions may be necessary to reduce risk. Unsanctioned SaaS applications either clearly provide no business benefits, or the security risks of the application outweigh the business benefits. For example, an unsanctioned SaaS application may violate regulatory compliance mandates, create an unacceptable risk of loss of corporate intellectual property or other sensitive data, or enable malware distribution (see Figure 3-15).

Figure 3-15

Impacts of sanctioned and unsanctioned SaaS applications



To control sanctioned SaaS usage, an enterprise security solution must provide the following:

- **Threat prevention.** SaaS applications introduce new threat risks that need to be understood and controlled. Many SaaS applications automatically sync files with users, and users often share data in SaaS applications with third parties that are out of an organization's control. These two aspects of SaaS environments create a new insertion point for malware that not only can get in from external shares but can also automatically sync those infected files across the organization without any user intervention. To address SaaS-based malware threats, a security solution must be able to prevent known and unknown malware from residing in sanctioned SaaS applications, regardless of the source.

- **Visibility and data exposure control.** After sanctioned SaaS usage is defined and controlled with a granular policy, data residing in those SaaS applications is no longer visible to the organization's perimeter firewalls. This loss of visibility creates a blind spot for IT. Additional data-exposure controls are needed to specifically address the unique risks associated with SaaS environments, with a focus on data protection. Visibility of data stored and used in SaaS applications is critical to ensuring a deep understanding of users, the data they have shared, and how they have shared it.

- **Risk prevention, not just risk response.** An organization's users commonly use certain SaaS applications long before the organization officially sanctions those applications. Even after a SaaS application is sanctioned, data is often shared with third parties that don't necessarily have next-generation security solutions to effectively safeguard SaaS data from malware threats and data-exposure risks. Threat prevention and data-exposure control in a SaaS-based environment require visibility and control not just from the time that a SaaS application is sanctioned going forward. You need visibility and control of *all* your data, including data that was being stored – and shared – before the SaaS application was sanctioned.

Data residing within enterprise-enabled SaaS applications is not visible to an organization's network perimeter. Prisma SaaS connects directly to sanctioned SaaS applications to provide data classification, sharing/permission visibility, and threat detection within the application. This capability yields unparalleled visibility, which allows organizations to inspect content for data-exposure violations and control access to shared data via a contextual policy.

Prisma SaaS builds on the existing SaaS visibility and granular control capabilities of the Security Operating Platform provided through App-ID, with detailed SaaS-based reporting and granular control of SaaS usage. Figure 3-16 shows an example of the granular controls for SaaS applications supported by App-ID.

Figure 3-16

Example of granular controls supported by App-ID

Application	Control	Feature
Box	Box – Personal	App-ID
	Box – Corporate	App-ID
	Upload control	File Blocking
	Download control	File Blocking
	Malware detection	WildFire & protection profile
	User-based control	User-ID

Prisma SaaS is a completely cloud-based, end-to-end security solution that provides visibility and control within SaaS applications, without the need for any proxies, agents, software, additional hardware, or network changes. Prisma SaaS isn't an inline service, so it doesn't impact latency, bandwidth, or end-user experience. Prisma SaaS communicates directly with the SaaS applications themselves and looks at data from any source, regardless of the device or location from which the data was sent.

3.4.3.1 SaaS threat prevention

WildFire (discussed in Section 2.6.2.4) integration with Prisma SaaS provides cyberthreat prevention to block known malware and to identify and block unknown malware. This integration extends the existing integration of WildFire to prevent threats from spreading through the sanctioned SaaS applications, which prevents a new insertion point for malware. When new malware is discovered by Prisma SaaS, the threat information is shared with the rest of the Security Operating Platform, even if it is not deployed inline with the SaaS applications.

3.4.3.2 Data-exposure visibility

Prisma SaaS provides complete visibility across all user, folder, and file activity, which provides detailed analysis that helps you transition from a position of speculation to one of knowing exactly what is occurring in the SaaS environment at any given point in time. Because you can view deep analytics of day-to-day usage, you can quickly determine whether there are any data-risk- or compliance-related policy violations. This detailed analysis of user and data activity allows for granular data governance and forensics.

Prisma SaaS connects directly to the applications themselves, so it provides continuous silent monitoring of the risks within the sanctioned SaaS applications, with detailed visibility that is not possible with traditional security solutions.

3.4.3.3 Contextual data-exposure control

Prisma SaaS enables you to define granular, context-aware policy control that provides you with the ability to drive enforcement and quarantine users and data as soon as a violation occurs. This control enables you to quickly and easily satisfy data-risk compliance requirements such as PCI and PII while still maintaining the benefits of cloud-based applications.

Prisma SaaS prevents data exposure in unstructured (hosted files) and structured (application entries such as Salesforce.com) data. Both data types are common sources of improper data shares.

3.4.3.4 Advanced document classification

Prisma SaaS inspects documents for common sensitive data strings (such as credit card numbers, SSH keys, and Social Security numbers) and flags them as risks if they are improperly shared. Unique to Prisma SaaS is the ability to identify documents by type through advanced document classification, regardless of the data that is contained in the document itself. Prisma SaaS has been designed to automatically identify sensitive documents, such as those related to medical, tax, and legal issues.

3.4.3.5 Retroactive policy

A traditional network security solution can see only inline data and apply security policies to data that is accessed inline, after the policy is created. This approach doesn't effectively prevent SaaS data exposure, however, because SaaS data may have been shared long before the policy was created. This data may not be accessed inline for many months or years, potentially leaving sensitive data exposed indefinitely to malware infection and unauthorized access.

Prisma SaaS retroactively applies security policies to all users and data from the beginning of the SaaS account's creation, rather than the policy's creation, to identify any potential vulnerabilities or policy violations. Prisma SaaS does not wait for someone to access the data inline to apply policies and resolve any vulnerabilities or violations; SaaS data and shares are proactively discovered, protected, and resolved, no matter when they were created.

Policies are context-driven to allow for granular definitions of data exposure risks. This granularity is necessary to enable SaaS use by users while still preventing accidental data exposure. Policies account for several factors in context to create an overall data-exposure risk profile. One or two factors may not provide enough insight into the potential risk of the share. The overall risk of exposure is determined only after the full context of the share is understood.

Risks are calculated by user type, document type, sensitive data contained, how the data is shared, and whether malware is present. This capability provides the ability to control the exposure at a granular level based on several important factors.

For example, a financial team may be able to share financial data with other people on their team, but not beyond that. Even though the original share is allowed, they cannot share data that is infected with malware. The financial team may, however, be allowed to share nonsensitive data company-wide or, in some cases, with external vendors. The key to enabling this level of granularity is the ability to look at the share in the context of all the factors.

3.4 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Fill in the Blank.** _____ provides continuous monitoring of public clouds and helps organizations achieve a continuous state of compliance in their public cloud workloads.
2. **Short Answer.** What are some of the organizational security risks associated with unsanctioned SaaS application usage?
3. **Short Answer.** Explain why traditional perimeter-based firewalls are not effective for protecting data in SaaS environments.
4. **True or False.** Prisma SaaS is deployed as a standalone inline service between the organization's traditional perimeter-based firewalls and requires a software agent to be installed on mobile devices.
5. **True or False.** Prisma SaaS protects data in hosted files and application entries.

3.5 Prisma Cloud Security Posture Management (CSPM)

Effective cloud security requires complete visibility into every deployed resource as well as absolute confidence in their configuration and compliance status. As enterprises further adopt cloud-native methodologies and gain the flexibility of multicloud architectures, stitching together security data from disparate legacy tools becomes a considerable obstacle. DevOps and security teams need a single, integrated solution like Prisma Cloud.

Prisma Cloud takes a unique approach to cloud security posture management, going beyond mere compliance or configuration management. Vulnerability intelligence from more than 30 data sources provides immediate clarity on critical security issues, while controls across the development pipeline prevent insecure configurations from ever reaching production. Prisma Cloud Security Posture Management (CSPM) modules include:

- **Visibility, Compliance, and Governance**
    - *Cloud Asset Inventory*. Prisma Cloud delivers comprehensive visibility and control over the security posture of every deployed resource. While some solutions simply aggregate asset data, Prisma Cloud analyzes and normalizes disparate data sources to provide unmatched risk clarity.

- *Compliance Monitoring and Reporting*. Prisma Cloud continuously monitors cloud compliance posture and supports one-click reporting from a single console. Numerous compliance frameworks are included out of the box, and you can build additional custom frameworks.

- *Infrastructure-as-code (IaC) Scanning*. Prisma Cloud enables users to scan IaC templates for vulnerabilities and build cloud-agnostic policies for the build and runtime development phases.

- **Threat Detection**

  - *User and Entity Behavior Analytics (UEBA)*. Prisma Cloud analyzes millions of audit events and then uses machine learning to detect anomalous activities that could signal account compromises, insider threats, stolen access keys, and other potentially malicious user activities.

  - *Network Anomaly Detection*. Prisma Cloud monitors cloud environments for unusual network behavior and can detect unusual server port or protocol activity, including port scan and port sweep activities that probe a server or host for open ports.

  - *Automated Investigation and Response*. Prisma Cloud provides automated remediation, detailed forensics, and correlation capabilities. Insights combined from workloads, networks, user activity, data, and configurations accelerate incident investigation and response.

- **Data Security**

  - *Data Visibility and Classification*. Prisma Cloud provides complete visibility into all Amazon Web Services Simple Storage Service (AWS S3) buckets and objects, including contents by region, owner, and exposure level. You can fine-tune data identifiers—such as driver's license, Social Security number, credit card number, or other patterns—to identify and monitor sensitive content.

  - *Data Governance*. Prisma Cloud includes specific data policies to quickly determine your risk profile based on data classification and exposure/file types. Enable or disable data compliance assessment profiles—for example, Payment Card Industry Data Security Standards (PCI DSS), General Data Protection Regulation (GDPR), System and Organization Controls Type 2 (SOC 2), and Health Insurance Portability and Accountability Act (HIPAA)—based on needs and generate audit-ready reports with a single click.

- *Malware Detection*. Prisma Cloud helps users identify and protect against known and unknown file-based threats that have infiltrated S3 buckets, leveraging the WildFire malware prevention service to flag any objects that contain malware.
- *Alerting and Remediation*. Prisma Cloud automatically generates alerts for each object based on data classification, data exposure, and file types. Analysts can take action on alerts to quickly remediate exposure, tag individual DevOps teams for violations, and delete any objects that contain malware.

Module 4 – Fundamentals of Security Operations

Knowledge Objectives

- Identify the key elements of security operations (SecOps) and describe SecOps processes.
- Describe SecOps infrastructure, including security information and event management (SIEM), analysis tools, and security operations center (SOC) engineering.
- Discuss security orchestration, automation, and response (SOAR) for SecOps.
- Identify the major components of the Cortex XDR deployment architecture and explain how it protects endpoints from malware and exploits.
- Describe how Cortex XSOAR, Cortex XSOAR TIM, and Cortex XSIAM deliver contextual threat intelligence to SOC teams to enable actionable insight into real-world attacks and automate security response actions.
- Explain how SOC teams can leverage Cortex Data Lake to collect, integrate, and normalize enterprise security data with advanced artificial intelligence (AI) and machine learning.

4.0 Elements of Security Operations

Security operations (SecOps) is a necessary function for protecting our digital way of life for our businesses and customers. SecOps requires continuous improvement in operations to face fast-evolving threats. SecOps needs to arm security operations professionals with high-fidelity intelligence, contextual data, and automated prevention workflows to quickly identify and respond to these threats. SecOps must leverage automation to reduce strain on analysts and execute the mission of the Security Operation Center (SOC) to identify, investigate, and mitigate threats. All of this, though necessary, can be very overwhelming for organizations building out a SecOps function or modernizing an existing SOC.

To increase confidence in the ability to quickly stop stealthy attacks and adapt defenses to prevent future attacks, a SecOps function requires the right set of building blocks. These building blocks include the people, process, and technology aspects required to support the business, the visibility that is required to defend the business, and the interfaces needed with other organizations outside of the SOC. By utilizing these elements to build a SecOps function, operations can be improved by increasing automation and accelerating investigations.

SecOps consists of six elements:

1. **Business** (goals and outcomes)
2. **People** (who will perform the work)
3. **Interfaces** (external functions to help achieve goals)
4. **Visibility** (information needed to accomplish goals)
5. **Technology** (capabilities needed to provide visibility and enable people)
6. **Processes** (tactical steps needed to execute on goals)

All of the elements tie back to the business itself and the goals of the SecOps organization.

Organizations use various delivery options for the SecOps function. The choice in delivery of security operations is usually driven by a few key factors, including the needs of the business, global presence, access to resources, and funding.

An in-house SOC keeps the knowledge and control of the environment within the business. However, depending on the size of the operation, it can require a considerable upfront investment. It is also dependent on being able to keep up with in-house staffing, which remains a major challenge for security organizations.

Outsourced security operations, or SOC-as-a-Service, provides access to experts, advanced technology, mature processes, and can be spun up quickly. However, it still requires in-house resources to carry out remediation tasks. It can also reduce the number of custom processes that can be put in place by the customer organization. It requires good service-level agreements (SLAs) and consistent monitoring and testing of the SLAs to ensure quality. This setup may also cause concerns around compliance at different global locations.

Many organizations choose a hybrid solution with some functions outsourced, such as level 1 analysts, to identify priority tasks. This solution provides access to additional skills that may not be present in-house and can provide both flexibility and scalability. It requires stringent interface agreements and tight processes around escalations and communication.

An emerging delivery option is to utilize robotic decision automation to fill the role of a level 1 analyst and then in-source or out-source the elevated tasks of a SOC. Often branded as artificial intelligence (AI), these systems utilize advanced probabilistic mathematics that can be trained to reason like a SOC analyst and can identify high-fidelity events that warrant further investigation.

4.0.1 Business objectives

4.0.1.1 Mission

Developing, documenting, and socializing the mission statement for your security operations is one of the most important elements of the organization. It will define to you, and to the business, the purpose of the SOC. This should include the objectives of the security operations organization and the goals the organization is expected to achieve for the business.

Socializing the mission statement and getting buy-in from executives provides clear expectations and scope of what the security operations team is responsible for. Some mission statements include defending an organization, protecting assets, or enabling the business. Some are customer-focused, as with service providers. Others provide for openness, as with university systems. Each is unique; however, they do have some common properties. The mission statement should define what actions will be taken, how those actions will be executed, and what the results are to the business.

4.0.1.2 Governance

Governance is how to measure performance against the defined and socialized mission statement. It defines the rules and processes put in place to ensure proper operation of the organization. It can include principles, mandates, standards, enforcement criteria, and SLAs. Additionally, it will define how the security operations team will be managed and who is responsible for ensuring the team is meeting the mission of the business. This should include actions to ensure the mission objectives are met.

4.0.1.3 Planning

Planning includes details on how the security operations organization is going to achieve its goals. Main business drivers will need to be identified and documented. Other inclusions are vision, strategy, service scope, deliverables, responsibilities, accountability, operational hours, stakeholders, and a statement of success.

Planning should be done with a three-year vision, which ensures continuity of operations—even in times of rotating executives that may have execution variances—to provide the expected value to the business. An investment strategy is also part of planning. This not only includes technology purchases but automation goals and investment in people. Plans should tightly align to the business. If there is a large merger and acquisition (M&A) strategy or digital transformation to the cloud, for example, the investment plan should align to those initiatives. Finally, the investment plan should demonstrate a clear return on investment (ROI) and value add for projects that support security operations. ROI/value add may also be demonstrated

through cost avoidance (for example, avoiding compliance penalties by implementing a specific security control).

4.0.2 Business execution

4.0.2.1 Staffing

Staffing of a security operations organization includes the recruitment, screening, and selection process for analysts and other SOC staff. Staffing of security skills remains one of the biggest challenges of the security industry, with additional challenges existing for organizations located outside of major tech hubs. Organizations with these issues should look to in-sourcing resources (analyst-as-a-service) to alleviate the strain of staffing.

Considerations must be made for the staffing model chosen (24x7, 8x5, co-sourcing, and others). On-call staffing requirements should be defined for critical incidents as well as out-of-hours support that is required. These considerations will drive the number of full-time employees (FTEs) required to meet the objectives of the team.

When building a security operations team, the most important hire is the security operations manager. They are responsible for creating the "people" procedures, managing the SOC staff, and working with other teams within the organization. For analyst hires, a general rule of thumb is two Tier 1 analysts for every Tier 2 analyst.

Organizations should avoid filling their team with "heroes." These types are valuable to an organization because they can (and do) perform all tasks but in the event of their departure, they are difficult and expensive to backfill. This presents a risk to the business that should be avoided. Instead, you want to staff the appropriate level of knowledge for each role in the SOC. There should be diversification of skills within the security operations organization such as malware analysis, network architecture, and threat intelligence; however, there should be basic knowledge and skill that has overlap amongst team members for vacation backup, illness, and attrition. Attrition will happen, so a healthy hiring pipeline must exist. This pipeline can come from internal hires from other parts of the organization (help desk, IT operations), universities, or from rotating staff throughout the business.

4.0.2.2 Career path progression

Retaining staff is also important and providing a clear career path is necessary to achieve this. A role's definition and skills matrix should be created, and a maintenance plan established in order to keep the skills matrix up to date. A semi-annual review of this content is suggested. Additionally, skills gaps/deficiencies should be continually updated by the SOC manager to allow visibility into possible capability holes to drive improvement.

Keep in mind that moving up to management is not the goal of all employees. The career path should allow for a management path, as well as a technical path. These paths should be documented with details of each job role and shared with the team, so they are aware of what is required at each level. Education opportunities should be available to help staff move through their preferred career path.

4.0.2.3 Employee utilization

Methods should be developed to maximize the efficiency of a security operations team specific to the existing staff. Security operations staff are prone to burnout due to console burn out and extreme workloads. To avoid this, team members should be assigned different tasks throughout the day. These tasks should be structured and may include:

- Shift turnover stand-up meeting (beginning of shift)
- Event triage
- Incident response
- Project work
- Training
- Reporting
- Shift turnover stand-up meeting (end of shift)

Another tactic to avoid burnout is to schedule shifts to avoid high-traffic commute times. Depending on the area, 8:00 a.m. to 5:00 p.m. may line up with peak (vehicle) traffic patterns. Shifting the schedule by two hours could reduce stress on the staff.

4.0.2.4 Training

Proper training of staff will create consistency within an organization. Consistency drives effectiveness and reduced risk. Having a formal training program will also enable the organization to bring on new staff quickly. Some organizations resort to on-the-job or shadow training for new hires, which is not recommended on its own. While shadowing other analysts during initial employment in the SOC is important, it should not be the only means of training.

Formal documentation should exist around capabilities, tools, processes, and communication plans (both internal and external) that new and existing staff can reference. Enablement plans for new tools should also be contained in the formal training program. This continuous education requires time and investment and should be supported by the business.

4.0.2.5 Facility

The facilities needed for your security operations team will depend on how you will be delivering the service—physically or virtually. A physical SOC may need separation from other parts of the business, including the Network Operations Center (NOC). While these two groups need to tightly interface with each other, they may need separate physical spaces to adhere to need-to-know principles and avoid specific legal issues. Where fusion centers are established, additional training for the network operations staff is required to ensure adherence to privacy principles.

The facility should include basic locking capabilities and preferably an advanced access schema that includes multi-factor authentication. Hand geometry readers with a PIN code are an example of advanced access control. Line-of-sight protection should also be accounted for. A closed/windowless room or snap glass is a way to achieve that protection. Snap glass can also lead to better morale since it lets in some outside light and reduces the "dungeon" feel that SOCs are typically known for.

Virtual SOCs (VSOC) are composed of team members that do not hold a physical space. They utilize online, secure portals to monitor traffic. When using a VSOC, extra care must be taken to secure the VPN and endpoint devices accessing the security portal, and a private space must be available for phone calls and discussions within the security operations team.

4.0.3 Business management and operations

4.0.3.1 Case management

A necessary capability for a SOC is a clear protocol for documenting and escalating incidents. Case management is a collaborative process that involves documenting, monitoring, tracking, and notifying the entire organization of security incidents and their current status. The minimum set of data points that should be captured in a case, and the tool selected for this function, should be capable of handling this data. Often, organizations will utilize multiple tools (ticketing, SOAR, email, and so on) for case management, which is ill-advised, as data continuity is severed, and incident handling efficiency takes a hit. Case management should also include a definition of who will have access to the data and tools, how cases will be documented in a consistent manner, and how teams will collaborate to close out incidents. A case management system should also be encrypted with strict access controls enforced due to the highly sensitive data that it will contain.

4.0.3.2 Budget

A financial plan for the costs of running the SOC should begin with an agreement on the mission of the SOC. Then, the technology, staff, facility, training, and additional needs to achieve that mission are identified. From there, a budget can be established to meet the team's minimum requirements. Often, a SOC budget is set from the top-down or assigned a percentage of an IT budget. This approach is not business focused and will result in frustration between capabilities and expectations from the business.

Once the budget is established, it should be followed by a regular review to identify additional needs or surplus. The timeline for regular budget requests and approval should be documented to avoid surprises or a last-minute rush to defend the organization's needs. Define the process needed to change the allocated budget, as well as a process for emergency budget relief.

A business-savvy budgeting resource can help the security operations organization navigate CapEx spending vs. OpEx spending and the expectations of the business. Be aware that government SOCs have additional considerations around the timing of elections and possible party-switching, which could result in dramatic budget shifts.

4.0.3.3 Metrics

If time is spent gathering metrics that cannot drive change, then they are, at best, a waste of time; at worst, these metrics can drive the wrong behavior. An example of this is Mean Time to Resolution (MTTR). This is a fine metric when used in a NOC (where uptime is key) but it can be detrimental when used in a SOC. Holding analysts accountable for MTTR will result in rushed and incomplete analyses. Analysts will rush to close incidents rather than do full investigations that can feed learning back into the controls to prevent future attacks. This will not produce better outcomes or reduced risk for the business.

Another poor metric is counting the number of firewall rules deployed. An organization may have ten thousand firewall rules in place, but if the first rule is "any-any" (allow traffic from any source to any destination), then the rest are useless. This is similar to measuring the number of data feeds into a SIEM. If there are 15 data feeds but only one use-case, then the data feeds aren't being utilized and are a potentially expensive waste.

Caution should be taken when measuring peoples' performance. Ranking top performers by number of incidents handled can have skewed results and may lead to analysts "cherry-picking" incidents that they know are fast to resolve. Additionally, evaluating individual performance in this way violates the law in various countries.

4.0.3.4 Reporting

Reporting is meant to give an account of what has been observed, heard, done, or investigated. It is to quantify activity and demonstrate the value the security operations team is providing to the business (or client organizations in the case of an MSSP). The outcome of reporting will not necessarily drive changes in behavior but is meant to track current activity. Reports are typically generated daily, weekly, and monthly.

Daily reports should include open incidents with details centered on daily activity. Weekly reports should identify security trends to initiate threat-hunting activities, which includes the number of cases opened and closed, and conclusions of the tickets (malicious, benign, or false positives). Include such information as how many different security use cases were triggered, their severity, and how they were distributed through hours of the day.

Monthly reports should focus on the overall effectiveness of the SecOps function. These reports should cover topics such as how long events are sitting in queue before being triaged, if the staffing in the SOC is appropriate (do more resources need to be added or reassigned?), what is the efficacy of rule fires, and are there rules that never fire or always fire a false-positive.

4.0.3.5 Business liaisons

A growing trend is for security organizations to hire business liaisons. This role is to tie-in to the different aspects of the business and help identify and explain the impact of security. This includes keeping up-to-date with new product launches and development schedules, onboarding new branch offices, and handling mergers and acquisitions where legacy networks/applications need to be brought into the main security program. This role can also be responsible for partner, vendor, and team interface management.

4.0.3.6 Governance, risk, and compliance

The governance, risk, and compliance (GRC) function is responsible for creating the guidelines to meet business objectives, manage risk, and meet compliance requirements. Common compliance standards are PCI-DSS, HIPAA, GDPR, etc. These standards require different levels of protection/encryption and data storage. Those requirements are typically handled by other groups; however, the breach disclosure requirements directly involve the security operations team. The SOC team must interface with the GRC team to define escalation intervals, contacts, documentation, and forensic requirements.

4.0.3.7 DevOps

The DevOps team is not only responsible for developing and implementing company-created applications, but also for maintaining them. This role has evolved greatly with the adoption of

cloud apps and agile development, where application upgrades are now rolled out within minutes, rather than the long cycles where we would see major releases every six to 12 months only. The DevOps team's main motivation is to push bug-free features out to users as rapidly as possible. Some groups have worked security protocols into their release cycles, but most have not.

Security operations will need to interface with the DevOps team to both work protocol into the release procedures and to get ahead of the new development tools and features that are being tested/used by DevOps. Additionally, the SecOps team will want to familiarize themselves with the DevOps processes and procedures in order to reduce friction between the teams.

4.0 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Fill in the Blanks.** The mission of the Security Operations Center (SOC) is to _____, _____, and _____ threats.
2. **Fill in the Blank.** _____ define the external functions that help the SOC achieve its goals.

4.1 Security Operations Processes

SecOps can be broadly defined as a function that identifies, investigates, and mitigates threats. If there is a person in an organization responsible for looking at security logs, then that fits the role of SecOps. Continuous improvement is also a key activity of a SecOps organization. The four main functions of SecOps are:

1. **Identify** – Identify an alert as potentially malicious and open an incident.
2. **Investigate** – Investigate the root cause and impact of the incident.
3. **Mitigate** – Stop the attack.
4. **Improve** – Adjust and improve operations to keep up with changing and emerging threats.

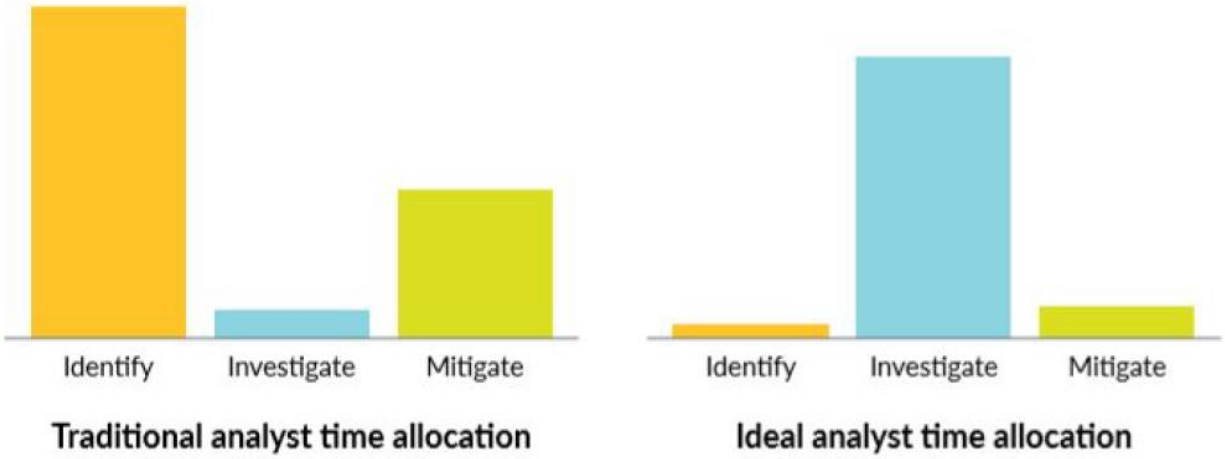
4.1.1 Identify

Security Operations Analysts spend the majority of their time in the identify phase due to false positives and low-fidelity alerts that they must weed through. Correctly implemented

prevention-based architectures and automated correlation help reduce the time needed for this phase. A lot of time is also spent in the mitigation phase. This is driven by the lack of automated remediation, combined with complex or deficient interfaces with teams outside of the security operations organization that need to be involved in halting the attack (see Figure 4-1).

Figure 4-1

The purpose of security operations is to identify, investigate, and mitigate threats.



4.1.1.1 Alerting

Alerting is how an event is determined to be important enough to become an actionable incident. The alerting function has a lot of opportunity to utilize automation. When automation is used to surface alerts, then it is crucial to define how these automated efforts will be validated for accuracy or missing events. The alerts themselves are generally created and maintained by the content engineering team. The quality of alerts is extremely important for presenting high-fidelity alerts to analysts and for reducing false positives.

4.1.1.2 Content engineering

Content engineering is the function that builds alerting profiles, which identify the alerts that will be forwarded for investigation. The content engineering team and the SecOps team need to be tightly interfaced, with feedback continuously flowing between the teams.

An interface agreement needs to be put in place identifying how often content updates will be made, how they will be vetted, and the feedback process. It should identify how the SecOps team and threat-hunting team make requests for new alerts or modifications to existing alerts. Properly configured alerts will allow the SecOps team to focus on important alerts that require further investigation.

4.1.1.3 Initial research

Initial research is a set of high-level processes that an organization utilizes to begin an investigation into a suspicious alert. The results of the initial research provide context around an incident to help in gathering information to triage, escalate, and determine if further investigation is needed or if the alert is malicious or benign.

When an alert is triggered, the SecOps team needs an easy way to gather the information required to determine the severity of an incident and build the foundation for an investigation. Many platforms offer automated severity recommendations and the data required for an analyst to quickly perform the initial review of alerts. Technology should be in place to allow for "right-click" or simple drill-down capabilities to access the context around the alert for the analyst to perform the initial research.

4.1.1.4 Severity triage

Severity triage defines the event prioritization based on impact to the business to help guide the analyst's actions through the incident-response lifecycle. When automation is utilized to assign an initial severity, the analyst reviews that severity assignment and then validates it against the uniqueness of the organization. They do this to verify or modify the severity and prioritization of the incident against other priorities.

Each business must determine its own risk tolerance and severity classifications. A 1-5 severity system is generally recommended: 1-Critical, 2-High, 3-Medium, 4-Low, and 5-Informational. "Critical" typically indicates a breach of some sort, although the exact descriptions and impacts will vary from business to business. Some organizations add a severity 0 to indicate an ongoing breach where the attacker is attempting to exfiltrate, encrypt, or corrupt data.

4.1.1.5 Escalation process

Escalation is a set of guidelines that enable the SecOps team to increase the organization's awareness of a potential issue and receive the necessary support. An interface agreement should be put in place to define what severities require increased awareness from the business. Escalation and communication plans need to be developed, documented, and socialized with all stakeholders. Stakeholders may include:

- IT Operations
- Governance, Risk Management, and Compliance (GRC)
- Legal
- Corporate Communications or Public Relations

- Support

An escalation matrix should be developed to include specific scenarios and the associated escalation steps. These plans can quickly become obsolete due to revolving staff, so regular reviews are necessary to ensure accuracy and relevancy. Backup contacts and procedures to address slow or inadequate responsiveness are recommended, as well.

4.1.1.6 Alert distribution

Alert distribution delegates analysts to work on specific alerts. Giving analysts the responsibility to work on a diverse set of alerts ensures that analysts have the opportunity to handle unfamiliar use cases and develop a well-rounded understanding of various alert types. Alert distribution challenges analysts to build their skillset and become familiar with their resources, and it alleviates the tendency an analyst may have to select and work on alerts they are familiar with. Working with diversified alert types prepares and equips analysts to respond to any given alert when needed.

4.1.2 Investigate

During the investigate phase, Security Operations Analysts perform a detailed analysis of an incident and collect forensics and telemetry data.

4.1.2.1 Detailed analysis

Detailed analysis is a deeper investigation into an incident to determine whether it is truly malicious, identify the scope of the attack, and document the observed impact. It is a manual process to answer the questions: what, where, when, why, who, and how. Additionally, a detailed analysis helps to confidently determine whether an incident is a true incident. In the event of a false positive, feedback should be provided to the content engineering team so they can tune alerts, or to the security engineering team so they can update controls as needed.

Detailed analysis is part of a modular incident-response plan. The analysis procedure presumes that initial research has concluded, and all respective pieces of information have been gathered accordingly. This procedure closes any remaining gaps that were left after the initial research. In addition, affected IT assets and business services are identified. The appropriateness and efficacy of available containment measures are evaluated and provided as input to the mitigation procedures. Detailed analysis ensures that all relevant information is gathered, including:

- The potential impact of the security incident
- The affected assets

- The adversary's objective
- The potential impact of containment measures

Only after these essential pieces of information have been investigated can the incident-response team make an informed decision about the containment and mitigation strategy.

4.1.2.2 Forensics and telemetry

Forensics and telemetry provide the data needed to perform the different types of investigation from severity triage to detailed analysis and hunting.

Telemetry is information about a broad range of activity gathered in real time from a given source. It is inclusive rather than selective, and it rarely collects the contents of an item. Examples of network telemetry would be session and packet headers, rather than packet contents. Endpoint telemetry would include process-execution details and file and memory reads and writes, but not their contents. Telemetry is consistently recorded, which makes it more useful than a log that collects prescribed information only when triggered by a specific event; it is also more accessible than forensics due to the wider coverage area and speed of collection.

Forensics is a commonly misused term, mostly referring to "the act of collecting raw data needed to complete the detailed analysis for an investigation." Raw data capture requires specific tools and tends to be slow due to its size and method of collection. In the case of network data, raw data would be capturing whole packets or netFlow logs; for endpoint data, it may include a memory dump, whole executable or operating-system files, or even whole hard drives. The true definition of the term "forensics" is the method that an expert witness uses to prepare evidence. For electronic (or computer) evidence to be admissible in a court of law, it must be repeatable and defensible; the process undertaken by an expert must not modify any of the original data in any way, and the results must be factual and not tainted by whichever party is funding the work. The term "forensics" defines this method, and raw data capture is an integral component.

Both telemetry and forensics are a necessity for every security team. Telemetry from network and endpoint activity, along with cloud configurations will provide readily available information necessary to triage and investigate the majority of alerts and incidents. Forensic data capture, while slow, will supplement telemetry and provide the information needed to conclude the small number of high-priority or difficult incident investigations that often lead to breach identification. Should a breach be validated, all data and results will be required by government and regulatory bodies; however, the forensic data will be of most use to their investigators because of the way it is collected and the depth of its contents.

Types of data include:

- **Event:** Any action performed by a person or technology
- **Alert:** Notification of an event
- **Log:** Details of an event
- **Telemetry:** Activity consistently gathered electronically and in real-time from a given source
- **Forensic (raw):** The complete contents of an item, without change or modification

4.1.3 Mitigate

Key processes in the mitigate phase include executing a mitigation strategy, performing preapproved mitigation scenarios, breach response, change control, and defining interface agreements.

4.1.3.1 Mitigation

Once an incident has been validated, a mitigation strategy must be executed. The mitigation strategy is comprised of a set of processes and interface agreements to contain the security incident. This typically includes documentation of any actions taken by the security team and temporary controls that can be implemented to quickly stop an attack, which should lead to permanent controls to prevent future attacks.

4.1.3.2 Preapproved mitigation scenarios

Some mitigation processes are easily automated for preapproved mitigation scenarios. These are a set of parameters that allow for the immediate containment or prevention of a security incident without further approvals. An example would be to block an infected laptop from the network to prevent the spread of malware. Another example is to create a dynamic process to block against specific indicators of compromise (IOCs) (such as known bad URLs, domains, or IP addresses) without requiring a security commit invoking a change window. The process that an analyst follows for each scenario, when executing the mitigation process, should be documented.

4.1.3.3 Breach response

A true breach requires a plan separate from standard mitigation. It defines how to effectively respond during a critical-severity incident. The first piece of this plan is to identify the cross-functional stakeholders, including corporate communications, legal teams, and third parties as

appropriate. Then assign a timeline of when each stakeholder should become involved and how they will initially be notified. Details of the information to be collected and shared by the SecOps team should be defined, as well as the SOC commander responsible for providing the information to the stakeholders. Also included should be information about the frequency of updates, method of updates, and communication processes (emails, collaboration tools, a war room, etc.). Training and policies should be created to prevent leaks of breach details beyond the breach-response team. Breach-response plans should be periodically tested, typically a few times per year, and at least once without the security team having prior knowledge of the test.

4.1.3.4 Change control

In cases of both manual and automated mitigation, a change-control process must be in place to monitor, document, and control changes being made. A good change-control process ensures that alterations to the environment have a minimized impact to business and are well documented in case a look-back review needs to be performed. The information required for this documentation should be identified and ideally contained in a formal template. This process should have timelines for reviewing and rolling back temporary changes. Also included, should be who can request changes, the steps needed to initiate change, any prerequisites or change windows available for the modification, backout and communications plans, and who can approve changes.

All changes should be:

- Deemed necessary to the business
- Consistently documented (even when automated)
- Planned and scheduled to minimize downtime or disruption

4.1.3.5 Interface agreements

Interface agreements define how the SecOps team and other teams will interact with each other. These agreements list the teams involved and detail the scope of work and responsibilities for each team. SLAs and operational-level agreements (OLAs) should be referred to, as well as change-request processes and escalation in cases where an interface agreement is not being upheld. Communication paths and tools used between the teams should be identified. A regular review of all agreements should occur, and the intervals of reviews set and stated clearly.

4.1.4 Improve

Continuous improvement includes tuning, process improvement, capability improvement, and quality review.

4.1.4.1 Tuning

Tuning refers to adjustments made to the alerting procedures regarding security incidents based on the outcomes of security investigations. It is an important step in reducing false positives and low-fidelity alerts in the SOC. An analyst may determine, during the course of a security incident, that there is a better way to detect the incident to increase visibility at the SIEM. When this occurs, the analyst will engage the tuning process to improve that visibility for future incidents. General tuning should be based on metrics collected from systems in the SOC. This includes a process to retire alerts when they are stale or ineffective.

The tuning process should define:

- Who or what triggers tuning efforts
- Thresholds for those triggers
- A review process for existing alerts
- The steps to request modifications to existing alerts (to increase visibility of future security incidents based on the outcome of a security investigation)

Alerts should be reviewed for tuning, at a minimum, on a quarterly basis alongside a monthly review of alert metrics.

4.1.4.2 Process improvement

Adjustments must also be made to the incident-response lifecycle based on the results of security incidents and new threats. New technologies introduced to the SOC and the business may also require incident-response process updates. The process should include information about who can update the incident-response processes (this person must be a qualified resource knowledgeable in incident response). Changes need not be made daily, so the process improvement plan should define how often processes should be reviewed, which will vary by process. All improvements should be reviewed and then operationalized and socialized with affected groups.

4.1.4.3 Capability improvement

Capability improvement is rooted in revisiting prior incidents and asking how these incidents can be better prevented or mitigated in the future. This results in adjustments to the alerting profile, prevention posture, and automation techniques. Sometimes the goal is to prevent an attack; at other times, it is to stop a breach faster or gather the appropriate information needed for quicker investigation. Ideally, this effort should be ongoing and follow every investigation. In most cases, this is not possible, so a monthly review of incidents should occur to identify opportunities for capability improvement.

Goals of capability improvement include:

- Prevention of future attacks
- Faster identification and stopping of a breach
- Gathering of necessary data for investigation of specific incidents
- Quicker investigations
- Automated remediation

4.1.4.4 Quality review

As new tuning measures, processes, and capabilities are implemented, a thorough peer evaluation of the changes should be carried out to ensure effectiveness and value to the business. Additionally, incident workflows and documentation should also be reviewed. This is to confirm consistency within the incident-response process which will result in a higher level of capability from the SecOps organization.

Organizations must document who is responsible for reviewing changes and closed cases, along with a cadence for the review process. This team must be given time to perform these reviews outside of their normal duties. A process should be created to define what severity cases require review, what items in the case will be reviewed, how feedback will be provided, and what training opportunities arise from the reviews. Then, that training must be delivered to the SecOps team (and sometimes beyond the SecOps team) to improve the overall efficiency and efficacy of preventing breaches.

4.1 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Short Answer.** Name the four main functions of SecOps.
2. **Multiple Choice.** Content engineering is an activity in which function of SecOps? (Choose one.)
    a) Identify
    b) Investigate
    c) Mitigate
    d) Improve
3. **Fill in the Blank.** _____ refers to adjustments made to the alerting procedures regarding security incidents based on the outcomes of security investigations.

4.2 Security Operations Infrastructure

Security Operations infrastructure includes a security information and event management (SIEM) platform, analysis tools, and SOC engineering.

4.2.1 Security information and event management

A SIEM platform, commercial or homegrown, is used as a central repository to ingest logs from all corporate-owned systems. SIEMs collect and process audit trails, activity logs, security alarms, telemetry, metadata, and other historical or observational data from a variety of different applications, systems, and networks in an enterprise. Most provide correlation capabilities, as well.

For a SIEM to operate properly, connectors and interfaces are required to ensure translated flow from the system of interest to the SIEM data lake. The SecOps organization should define how ownership of an event will be established, as well as the central point where an analyst will go to receive alerts. Sometimes it is the SIEM, in other cases, it is a security orchestration, automation, and response (SOAR) platform or ticketing system.

The selected SIEM approach should address any governance, risk, and compliance requirements for the separation of data, privacy, and retention times. This will drive requirements on storage

space and controls. Limiting data redundancy between the SIEM and feeder systems can help control costs, as well as controlling offline storage for long-term compliance needs.

4.2.2 Analysis tools

Analysis tools include advanced techniques, tools, and algorithms that provide the ability to detect evidence of security compromise within large volumes of data. Processes should be defined around how an analyst will determine whether an alert is malicious, and the chosen tools should assist or automate this process. Additionally, the tools should provide access to gather context about the given event, preferably in an automated way. Ownership, budget, and the support model for the tools needs to be defined.

Analysis tools are often based on machine learning, deep learning, and artificial intelligence, which provide either stand-alone, embedded, or add-on functionality to detect evidence of a security compromise. Security analytics can be performed on data that is either stored at rest or collected in motion—even at line speed on a massive network. This is a capability that SecOps teams can obtain in a variety of ways, with most security products and services including some sort of security analytics function.

4.2.3 SOC engineering

The SOC engineering team is responsible for the implementation and ongoing maintenance of the SecOps team's tools, including the SIEM and analysis tools. It is important to clearly define the responsibilities of this team. Will they be responsible for the licensing, maintenance, and updating of the tools? Will they manage the underlying architecture (CPU, RAM, storage, cloud implementation), or will that be handled by another team? SLAs with the team should be defined to cut down friction between teams, as well as to establish clear communication plans.

4.2 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Fill in the Blank.** A _____ collects and processes audit trails, activity logs, security alarms, telemetry, metadata, and other historical or observational data from a variety of different applications, systems, and networks in an enterprise.

4.3 Security Operations Automation

Security operations automation is achieved through security orchestration, automation, and response (SOAR) technologies and management of security automation functions.

4.3.1 Security orchestration, automation, and response

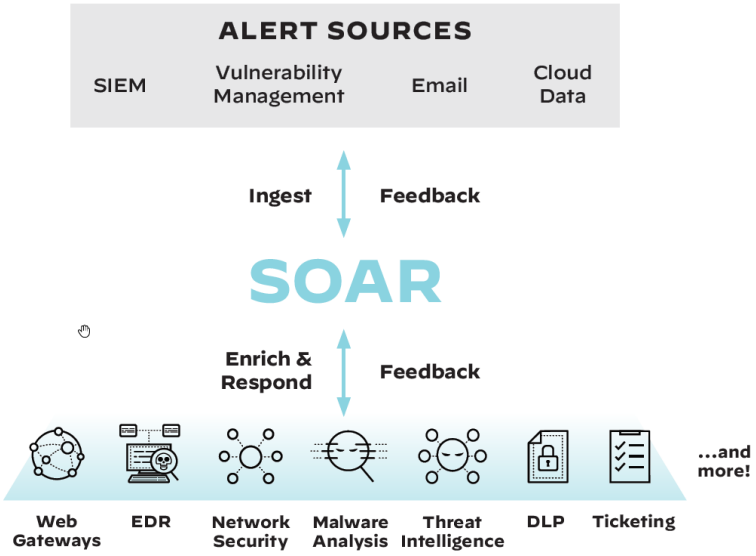
According to Gartner:

> SOAR refers to technologies that enable organizations to collect inputs monitored by the security operations team. For example, alerts from the SIEM system and other security technologies—where incident analysis and triage can be performed by leveraging a combination of human and machine power—help define, prioritize, and drive standardized incident response activities. SOAR tools allow an organization to define incident analysis and response procedures in a digital workflow format.[39]

SOAR systems allow for accelerated incident response through the execution of standardized and automated playbooks that work upon inputs from security technology and other data flows. SOAR tools ingest aggregated alerts from detection sources (such as SIEMs, network security tools, and mailboxes) before executing automatable, process-driven playbooks to enrich and respond to these alerts. The playbooks coordinate across technologies, security teams, and external users for centralized data visibility and action. They help accelerate incident response times and increase analyst productivity. By standardizing processes, they provide consistency, which improves operational confidence in SOC capabilities (see Figure 4-2).

Figure 4-2

High-level view of how SOAR tools sit in a SOC



[39] Gartner Glossary. (n.d.). *Security Orchestration, Automation and Response (SOAR)*. Gartner. Retrieved June 4, 2020, from https://www.gartner.com/it-glossary/security-orchestration-automation-response-soar

4.3.2 Security automation

Consistency is a key factor in the effectiveness of a SecOps team. Automation helps ensure consistency through machine-driven responses to security issues. A security automation function will own and maintain these automation tools. They must be tightly integrated with the SecOps team to continuously maintain the automation playbooks. They are also responsible for implementing new automation technology and playbooks in response to new workflows and processes defined by the SecOps team. The requirements, and eventual vetting of the solution, should be the responsibility of the SecOps teams. This vetting should consider the time savings, accuracy, and usefulness of the automation.

There are some cases in which automation increases the need for resources. It is always necessary to consider the return on investment (ROI) before investing in automation. When doing an ROI analysis, take special care to consider the ongoing cost of maintenance and support.

4.3 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **Fill in the Blank.** Technologies that enable organizations to collect inputs monitored by the security operations team are known as _____.

4.4 Secure the Future (Cortex)

Cortex is an artificial-intelligence-based, continuous security platform. Cortex allows organizations to create, deliver, and consume innovative new security products from any provider without additional complexity or infrastructure.

Security teams are constantly challenged to prevent data breaches. The issues originate from too many alerts, too few security analysts, narrowly focused tools, lack of integration, and lack of time. The more they react, the further behind they get. Palo Alto Networks has developed a breakthrough approach to SOC visibility, investigation, and speedy resolution called Cortex XDR. Cortex XDR brings visibility to the security team across all aspects of the infrastructure, breaking down silos and presenting a holistic picture of the organization's activity in order to improve security operations and posture.

From a business perspective, Cortex XDR enables organizations to prevent successful cyberattacks as well as simplify and strengthen security processes. This capability, in turn,

enables organizations to better serve users and accelerate digital transformation initiatives—because when users, data, and applications are protected, companies can focus on strategic priorities. With Cortex XDR, you can uncover stealthy threats with behavior analytics, investigate events, and hunt down threats with powerful search tools.

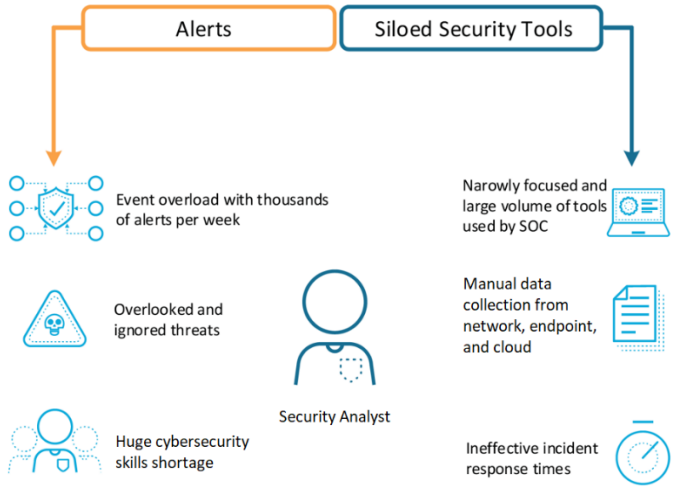
Security operations (SecOps) is a joint effort between IT teams such as security and operations working to prevent threats and detect and respond to security incidents. The goal of any security team is to defend an organization's infrastructure and data from damage, unauthorized access, and misuse. Larger organizations operate a security operations center (SOC), where a team of dedicated security staff detect, investigate, and respond to threats with tools to determine the extent of the threat through analysis and threat-hunting techniques.

For organizations, as threats continue to escalate in sophistication and numbers, SecOps is put under increased pressure by the large number of alerts that they receive, which makes it impossible to effectively deal with them. Another challenge that SecOps faces when dealing with all the alerts they receive is the lack of overall context for their investigations when dealing with multiple separate platforms that are generating alerts. As a result, the SecOps teams have to manually integrate multiple data sources and tools to understand the attack, causing investigations to take too long and potential threats to be missed.

Achieving 100 percent prevention is extremely difficult for any organization. Security operations centers (SOCs) today purchase many niche security products, which creates the disadvantage of having to track and manage so many alerts coming in from different platforms and tools. It can take days to weeks for one SOC engineer to investigate a single suspicious activity or alert, which may lead to nothing in the end (see Figure 4-3).

Figure 4-3

Struggles of a security analyst



Palo Alto Networks has a different approach for SecOps teams:

1. First, you prevent all of the threats you can with Cortex XDR endpoint protection and our next-generation firewalls.

2. Everything you can't prevent, you need to detect and investigate rapidly. You achieve this with Cortex XDR, CORTEX XSOAR TIM, and Cortex Data Lake.

3. Then you continuously automate responses with Cortex XSOAR. Cortex XSOAR allows security teams to ingest alerts across multiple sources and then execute automatable playbooks for accelerated incident response.

4. Finally, you enable real-time autonomous threat response with Cortex XSIAM. Cortex XSIAM is an AI-powered SOC platform that revolutionizes the way data, analytics, and automation get deployed to outpace modern threats.

Cortex is the platform for SecOps. Think of Cortex as your one-stop shop for SecOps, solving all key challenges in a more efficient way with better security outcomes. With Cortex, you can speed up investigations by having the right data—integrated across network, endpoint, and cloud—with all the context needed for security analysts.

4.4.1 Endpoint protection (Cortex XDR)

Adversary strategies have evolved from simple malware distribution to a broad set of automated, targeted, and sophisticated attacks that can bypass traditional endpoint protection. This evolution has forced organizations to deploy multiple products from different vendors to protect against, detect, and respond to these threats. Cortex XDR brings powerful endpoint protection together with endpoint detection and response (EDR) in a single agent. You can replace all your traditional antivirus agents with one lightweight agent that shields your endpoints from the most advanced adversaries by understanding and blocking all elements of attacks.

Although attacks have become more sophisticated and complex, they still use basic building blocks to compromise endpoints. The primary attack methods continue to exploit known and unknown application vulnerabilities as well as deploying malicious files, including ransomware. These attack methods can be used individually or in various combinations, but they are fundamentally different in nature:

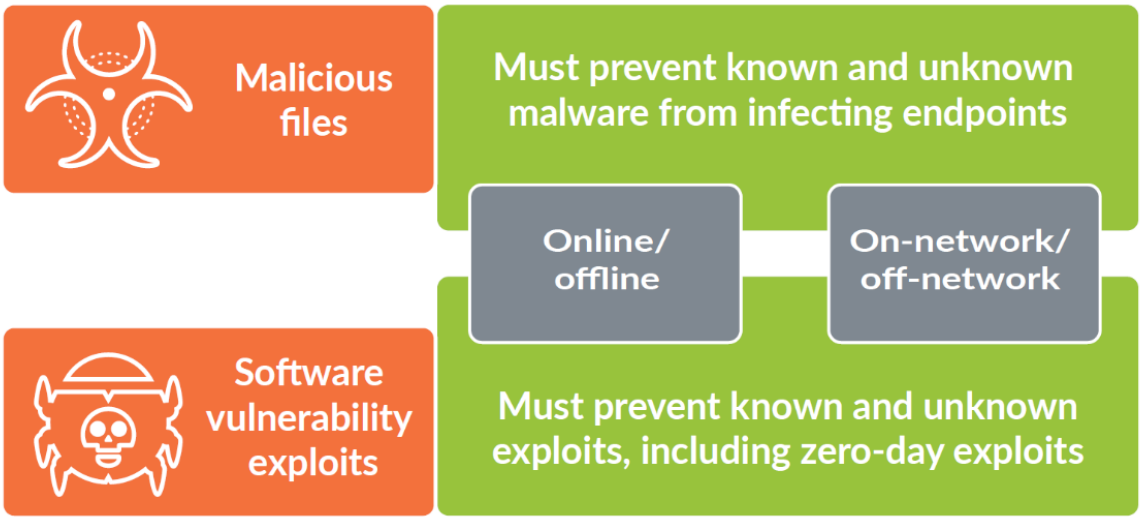
- Exploits are the results of techniques used against a system that are designed to gain access through vulnerabilities in the code of an operating system or application.

- Malware is a file or code that infects, explores, steals, or conducts virtually any behavior an attacker wants.
- Ransomware is a form of malware that holds valuable files, data, or information for ransom, often by encrypting data, with the attacker holding the decryption key.

Due to the fundamental differences between malware and exploits, effective prevention must protect against both. The Cortex XDR agent combines multiple methods of prevention at critical phases within the attack lifecycle to halt the execution of malicious programs and stop the exploitation of legitimate applications, regardless of operating system, the endpoint's online or offline status, or whether the endpoint is connected to an organization's network or roaming (see Figure 4-4).

Figure 4-4

Malicious files versus exploits

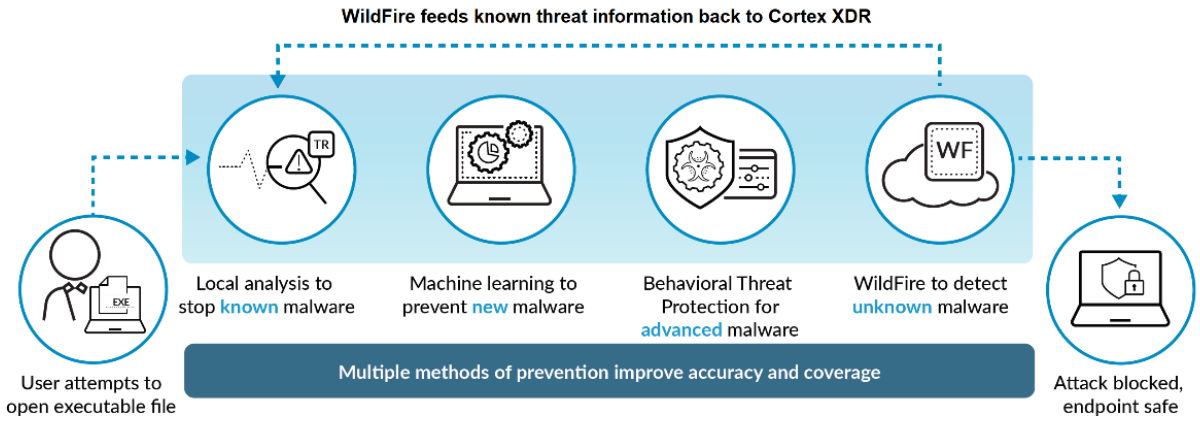


4.4.1.1 Stop malware and ransomware

The Cortex XDR agent prevents the execution of malicious files with an approach tailored to combating both traditional and modern attacks. Additionally, administrators can use periodic scanning to identify dormant threats, comply with regulatory requirements, and accelerate incident response with endpoint context. The Cortex XDR agent also performs scheduled or on-demand scans for dormant malware in malicious Microsoft Office files with macros, executable files, and dynamic link libraries (DLLs) to remediate these without the malicious files being opened. Known and unknown malware, including ransomware, is subject to multiple preventive technologies within Cortex XDR (see Figure 4-5).

Figure 4-5

Cortex XDR leverages multiple technologies and techniques to protect endpoints from known and unknown malware.



WildFire threat intelligence

In addition to third-party feeds, Cortex XDR uses the intelligence obtained from tens of thousands of subscribers to the Palo Alto Networks WildFire malware prevention service to continuously aggregate threat data and maintain the collective immunity of all users across endpoints, networks, and cloud applications.

1. Before a file runs, the Cortex XDR agent queries WildFire with the hash of any Windows, macOS, or Linux executable file, as well as any dynamic link library (DLL) or Office macro, to assess its standing within the global threat community. WildFire returns a near-instantaneous verdict on whether a file is malicious or benign.

2. If a file is unknown, the Cortex XDR agent proceeds with additional prevention techniques to determine whether it is a threat that should be blocked.

3. If a file is deemed malicious, the Cortex XDR agent automatically terminates the process and (optionally) quarantines the file.

Local analysis and machine learning

If a file remains unknown after the initial hash lookup, the Cortex XDR agent uses local analysis via machine learning on the endpoint—trained by the rich threat intelligence from global sources, including WildFire—to determine whether the file should run. By examining thousands of file characteristics in real time, local analysis can determine whether a file is likely malicious or benign without relying on signatures, scanning, or behavioral analysis. The model is built on a

unique agile framework, enabling continuous updates to ensure that the latest local prevention is always available.

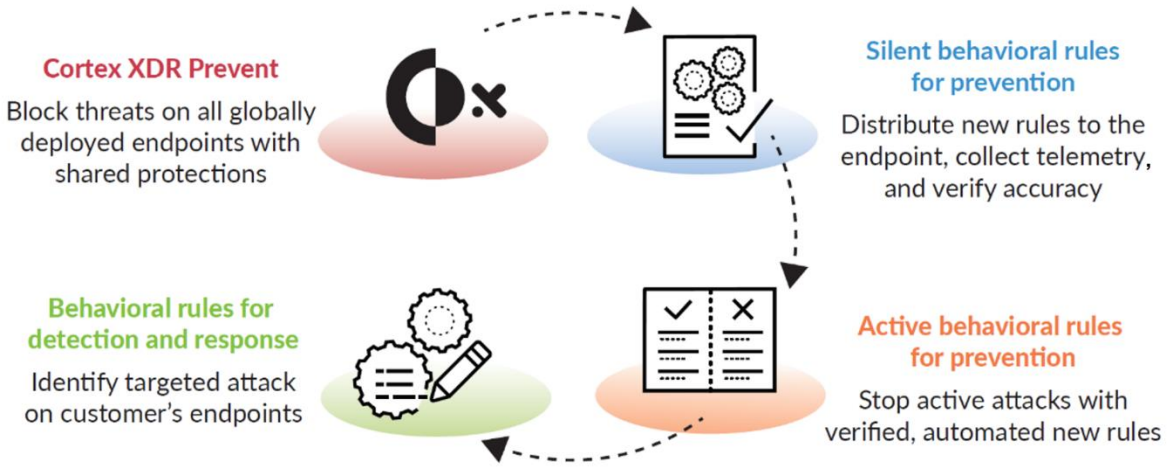
Behavioral threat protection

Sophisticated attacks that use multiple legitimate applications and processes for malicious operations have become more common, are hard to detect, and require deeper visibility to correlate malicious behavior. For behavior-based protection to be effective, including identification of malicious activity occurring within legitimate processes, it's critical to understand everything happening on the endpoint. The Cortex XDR agent enacts behavior-based protection in a few different ways.

Endpoint attacks often comprise multiple events that occur in the system. By itself, each event appears benign because attackers use legitimate applications and operating-system functions to achieve their goal. Strung together, however, they may represent a malicious event flow. With behavioral threat protection, the Cortex XDR agent can detect and act on malicious chains of events that target multiple operations on an endpoint, such as network, process, file, and registry activity (see Figure 4-6). When the Cortex XDR agent detects a match, it executes a policy-based action, such as "block" or "alert." In addition, it reports the behavior of the entire event chain up to the console and identifies the actor that caused the activity chain. The Cortex XDR agent can also quarantine files that were involved in malicious flows. Behavioral threat protection is ideal for protecting against script-based and fileless attacks.

Figure 4-6

Behavioral threat protection with Cortex XDR



The granular *child process* protection module prevents script-based attacks used to deliver malware by blocking known targeted processes from launching child processes that are commonly used to bypass traditional security approaches. The Cortex XDR agent prevents script-based and file-less attacks by default with out-of-the-box, fine-grained controls over the launching of legitimate applications, such as script engines and command shells; it also continues to expand these controls through regular content updates. Administrators have additional flexibility and control; they can allow list or block list child processes, and use command-line comparisons, to increase detection without negatively affecting process performance or shutting processes down.

Key Terms

In multitasking operating systems, a *child process* is a subprocess created by a parent process that is currently running on the system.

The behavior-based ransomware protection module protects against encryption-based behavior associated with ransomware by analyzing and stopping ransomware activity before any data loss occurs. To combat these attacks, Cortex XDR employs decoy files to attract the ransomware. When the ransomware attempts to write to, rename, move, delete, or encrypt the decoy files, the Cortex XDR agent analyzes the behavior and prevents the ransomware from encrypting and holding files hostage. When configured to operate in prevention mode, the Cortex XDR agent blocks the process attempting to manipulate the decoy files. When you configure this module in notification mode, the agent logs a security event.

WildFire inspection and analysis

In addition to local analysis, Cortex XDR can send unknown files to WildFire for discovery and deeper analysis to rapidly detect potentially unknown malware. WildFire brings together the benefits of independent detection techniques for high-fidelity and evasion-resistant discovery that goes beyond legacy approaches. These techniques include:

- Static analysis is a powerful form of analysis, based in the cloud, that detects known threats by analyzing the characteristics of samples before execution.
- Dynamic analysis (sandboxing) detonates previously unknown submissions in a custom-built, evasion-resistant virtual environment to determine real-world effects and behavior.

- Bare-metal analysis uses a hardware-based analysis environment specifically designed for advanced threats that exhibit highly evasive characteristics and that can detect virtual analysis.

If WildFire determines a file to be a threat, it automatically creates and shares a new prevention control with the Cortex XDR agent and other Palo Alto Networks products in minutes to ensure that the threat is immediately classified as malicious and blocked if it is encountered again.

4.4.1.2 Block exploits and fileless threats

Rather than relying on signatures or behavior-based detection to identify exploit-based attacks, the Cortex XDR agent takes the unique approach of targeting the limited set of techniques, or tools, any exploit-based attack must use to manipulate a software vulnerability. By preventing the use of these techniques—instead of identifying each individual attack—the Cortex XDR agent uses multiple methods to prevent zero-day exploits and protect unpatched systems, shadow IT (applications IT is unaware of), and unsupported legacy systems.

Pre-exploit protection

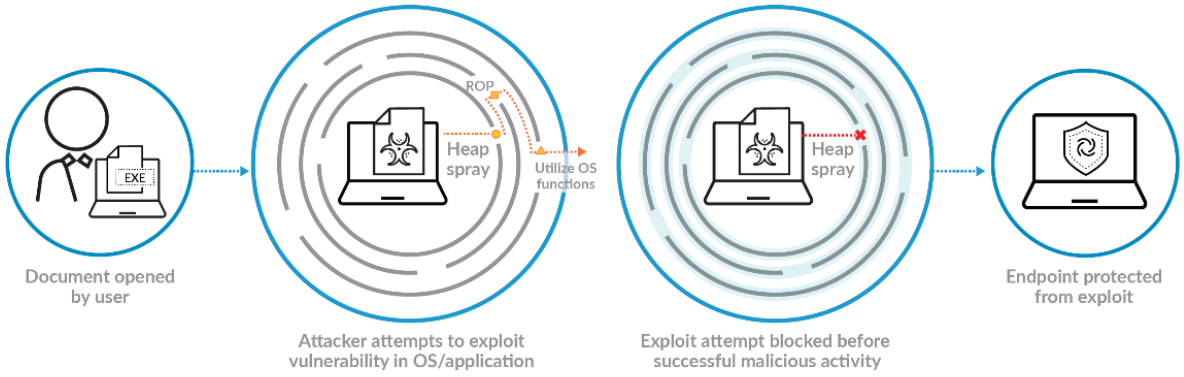
The Cortex XDR agent prevents the vulnerability-profiling techniques that exploit kits use before launching attacks. By blocking these techniques, the agent prevents attackers from targeting vulnerable endpoints and applications, effectively stopping the attacks before they begin.

Technique-based exploit prevention

The Cortex XDR agent prevents known, zero-day, and unpatched vulnerabilities by blocking the exploitation techniques that attackers use to manipulate applications (see Figure 4-7). Although there are thousands of exploits, they typically rely on a small set of exploitation techniques that change infrequently. By blocking these, Cortex XDR prevents exploitation attempts before endpoints can be compromised.

Figure 4-7

Cortex XDR focuses on exploit techniques rather than on the exploits themselves.



Kernel exploit prevention

The Cortex XDR agent prevents exploits that use vulnerabilities in the operating-system kernel to create processes with escalated, system-level privileges. It also protects against new exploit techniques used to execute malicious payloads, such as those seen in the 2017 WannaCry and NotPetya attacks. By blocking processes from accessing the injected malicious code from the kernel, the Cortex XDR agent can stop an attack early in the attack lifecycle without affecting legitimate processes. This capability enables the agent to block advanced attacks that target or stem from the operating system itself.

By blocking the techniques common to exploit-based attacks, the Cortex XDR agent allows you to:

- **Protect applications that can't be patched and shadow IT applications.** The Cortex XDR agent enables organizations to run any applications—including those developed in-house, those no longer receiving updates or security support, or those running in the environment without IT's awareness—without opening the network to the threat of exploit-based attacks.
- **Prevent successful zero-day exploits.** Because the Cortex XDR agent blocks the limited set of exploitation techniques that zero-day exploits typically use, it protects organizations against attacks that utilize zero-day exploits.
- **Eliminate the need to urgently patch applications.** Organizations using the Cortex XDR agent can apply security patches when it is best for the business and after sufficient testing. It prevents the exploitation of application vulnerabilities regardless of when an organization applies security patches issued by application vendors.

Credential-theft protection

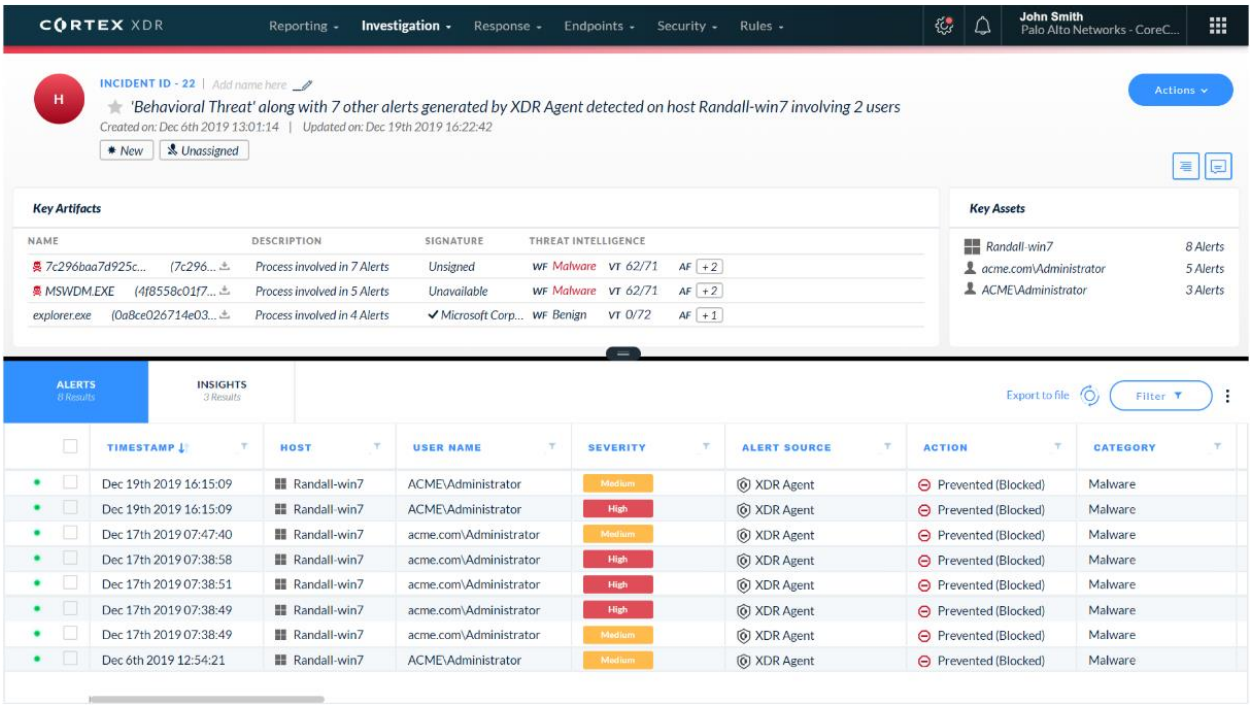
Attackers steal credentials to impersonate valid users, move uninterrupted through targeted organizations' networks, and find and exfiltrate valuable data. The Cortex XDR agent prevents credential theft tools like Mimikatz from accessing system passwords, ensuring that adversaries and malicious insiders cannot misuse credentials or escalate privileges. For additional credential-theft protection, Cortex XDR can collect endpoint events, profile behavior, and detect credential-based attacks to eliminate hard-to-find attacks.

4.4.1.3 Investigate and respond to attacks

To facilitate faster investigation and response, Cortex XDR offers administrators and incident-response teams multiple means to further their investigations, collect necessary information, and make any necessary changes to the endpoint in question (see Figure 4-8).

Figure 4-8

Investigate and respond to attacks.



When remediation on the endpoint is needed following an alert or investigation, administrators have the option to take the following actions:

- **Isolate endpoints** by disabling all network access on compromised endpoints except for traffic to the Cortex XDR management console, preventing these endpoints from communicating with and potentially infecting other endpoints.

- **Terminate processes** to stop any running malware from continuing to perform malicious activity on the endpoint.
- **Block additional executions** of a given file by block listing it in the policy.
- **Quarantine malicious files** and remove them from their working directories if the Cortex XDR agent has not already quarantined the files.
- **Retrieve specific files** from endpoints under investigation for further analysis.
- **Directly access endpoints with Live Terminal**, gaining the most flexible response actions in the industry to run Python, PowerShell, or system commands or scripts; review and manage active processes; and view, delete, move, or download files.
- **Orchestrate response with open APIs** that allow third-party tools to apply enforcement policies and collect agent information from any location.

4.4.1.4 Extending prevention beyond Windows environments

Although native security has grown among major operating-system vendors, such security remains focused on their own OS, creating fragmented protection, policies, enforcement, and visibility. Organizations need to be able to apply security rules across a mixed environment from a single screen as well as protect against a range of threats, from basic to advanced.

Through the Cortex XDR console, organizations can control default and custom security policies across Windows, macOS, Linux, and Android endpoints with confidence that multiple methods of protection are keeping their systems safe from attack.

Cortex XDR for macOS

Cortex XDR secures macOS systems against malware and exploits with more than just “check box” security. The Cortex XDR agent uses multiple methods—such as local analysis, WildFire inspection and analysis, Gatekeeper enhancements, trusted publisher identification, and administrator override policies—to block malware. To prevent exploits, the agent blocks kernel privilege escalation and exploitation techniques, including *JIT, ROP, and dylib hijacking*.

The Cortex XDR agent prevents attackers from bypassing the macOS digital-signature verification mechanism, Gatekeeper. This mechanism allows or blocks the execution of applications based on their digital signatures, which are ranked in three signature levels: Apple System, Mac App Store, and Developers. It extends Gatekeeper functionality to enable customers to specify whether to block all child processes or to allow only those with signature levels that match or exceed those of their parent processes.

Cortex XDR for Android

The Cortex XDR agent prevents known malware and unknown *Android Package Kit (APK)* files from running on Android endpoints. It enforces your organization's security policy as defined in the Cortex XDR console. The security policy determines whether to block known malware and unknown files, upload unknown files for in-depth inspection and analysis, treat malware as grayware, or perform local analysis to determine the likelihood that unknown files are malware. You can also allow list trusted signers to enable unknown, signed apps to run before the Cortex XDR agent receives an official verdict for the app.

Key Terms

A *Just-in-Time (JIT)* exploit targets a client-side JIT compiler used in Java programming.

A *return-oriented programming (ROP)* exploit allows an attacker to take control of a program by executing code that diverts the execution flow of the program.

A *dylib hijacking* exploit injects malicious code into an application that searches for and loads dynamic libraries in a vulnerable manner.

An *Android Package Kit* (APK) file is an app created for the Android mobile operating system.

Cortex XDR for Linux

The Cortex XDR agent protects Linux servers by preventing attackers from executing malicious ELF files or exploiting known or unknown Linux vulnerabilities to compromise endpoints. The agent also extends protection to processes that run in Linux containers. The Cortex XDR agent enforces your organization's security policy as defined in the Cortex XDR console. When a security event occurs on your Linux server, the Cortex XDR agent collects forensic information that you can use to analyze the incident further. The Cortex XDR agent on Linux operates transparently in the background as a system process. When you install it on a Linux server, it automatically protects any new or existing containerized processes regardless of how the container is deployed or managed.

Device control for secure USB access

USB devices offer a variety of benefits, but they also introduce risk. When users unwittingly connect malware-laden flash drives to their computers or copy confidential data to backup disk drives, they expose their organizations to attack and data loss. Advanced attackers can even infect seemingly innocuous USB devices such as keyboards and webcams with malware. The powerful device control module included with Cortex XDR allows you to monitor and secure

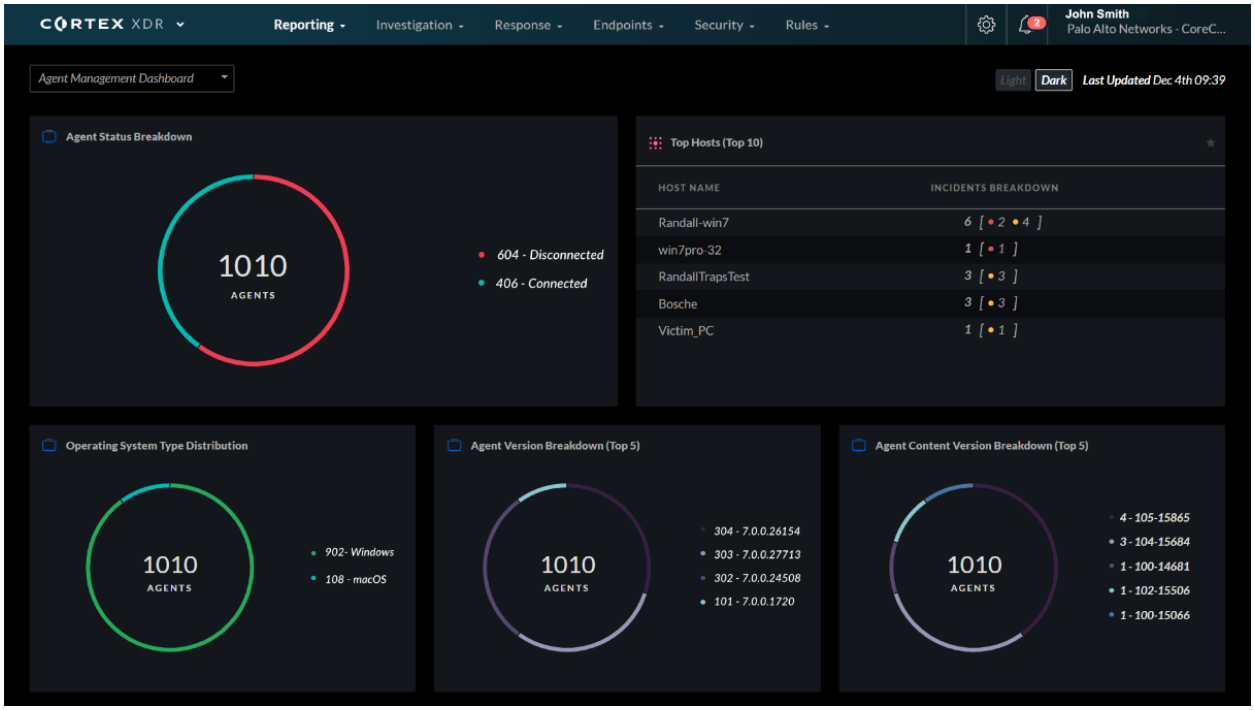
USB access without needing to install another endpoint agent on all your hosts. You can assign policies based on Active Directory group and organizational unit, restrict usage by device type, and assign read-only or read/write policy exceptions by vendor, product, and serial number. The device control module allows you to easily manage USB access and gain peace of mind that you've mitigated USB-based threats.

4.4.1.5 Simple endpoint security management

With an intuitive, web-based user interface (see Figure 4-9), Cortex XDR helps administrators quickly coordinate and protect your organization with out-of-the-box, day-one capabilities, without sacrificing your complex environment's need for control and customization.

Figure 4-9

The Cortex XDR dashboard



Cloud-based management

The multiregion, cloud-based Cortex XDR service saves you from investing in building out your own global security infrastructure and ties in to the suite of Palo Alto Networks products for additional integration and value. The service is simple to deploy and requires no server licenses, databases, or other infrastructure to get started, enabling your organization to protect hundreds or millions of endpoints without incurring additional operating costs.

Intuitive interface

Cortex XDR was designed to address security teams' growing responsibilities with an interface that makes it easy to manage policies and events as well as accelerate incident response. By combining endpoint policy management, detection, investigation, and response in one web-based management console, Cortex XDR provides a seamless platform experience. You can quickly assess security status with customizable dashboards and summarize incidents and security trends with graphical reports that can be scheduled or generated on demand. You can also deploy and upgrade Cortex XDR agents easily from a central location.

Elements of the interface include:

- **Multiple grouping methods**, including static groups or dynamic groups. Dynamic grouping can be based on endpoint characteristics such as a partial hostname or alias, full or partial domain or workgroup name, IP address, range or subnet, installation type such as virtual desktop infrastructure (VDI), agent version, endpoint type, or operating-system version.
- **Security profiles and simplified, rule-based policies** to protect endpoints out of the box while enabling granular customization for sensitive departments or individuals and easy reuse of settings across different endpoint groups.
- **Incident management** to help identify high-priority events and enable teams to communicate on status, progress, and other useful information. Integrated WildFire analysis displays information such as hash values, targeted users, applications, processes, and URLs involved in delivery or phone-home activities for incident response.

4.4.1.6 Benefits of a connected platform

By tightly integrating with the Palo Alto Networks suite of products, the Cortex XDR agent continuously exchanges threat information and data with WildFire—and endpoint incident and event logs with Cortex Data Lake, a cloud-based data collection, storage, and analysis service—to help your organization coordinate and automate enforcement across your entire security ecosystem, including endpoints, networks, and clouds.

Native integration for fast investigation and response

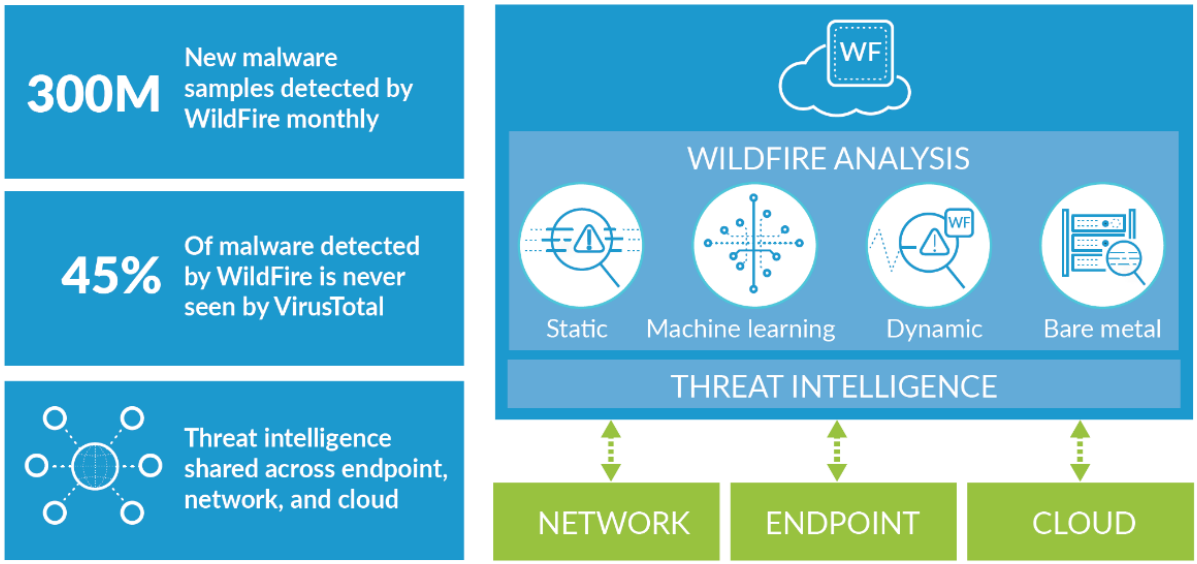
The data collected from the Cortex XDR agent is stored in Cortex Data Lake, which delivers efficient log storage that scales to handle the large volume of data needed for analytics, detection, and response. You can quickly deploy Cortex XDR and Cortex Data Lake, avoiding the time-consuming process of setting up new equipment.

By eliminating on-premises log storage and additional sensors and enforcement points, Cortex XDR can reduce total cost of ownership by an average of 44 percent. Cortex XDR also boosts the productivity of your security operations team by automatically detecting attacks and accelerating investigations.

Cortex XDR is the world's first detection and response app that breaks silos by natively integrating endpoint, cloud, and network apps to stop sophisticated attacks. Cortex supports data from Palo Alto Networks next-generation firewalls, Prisma Access, and Cortex XDR agents, in addition to third-party alerts and logs (see Figure 4-10). It speeds alert triage and incident response by providing a complete picture of an attack, including root cause, and stitching together the sequence of events to simplify investigations. Intelligent alert grouping and deduplication reduce the number of individual alerts to review by 98 percent, alleviating alert fatigue.

Figure 4-10

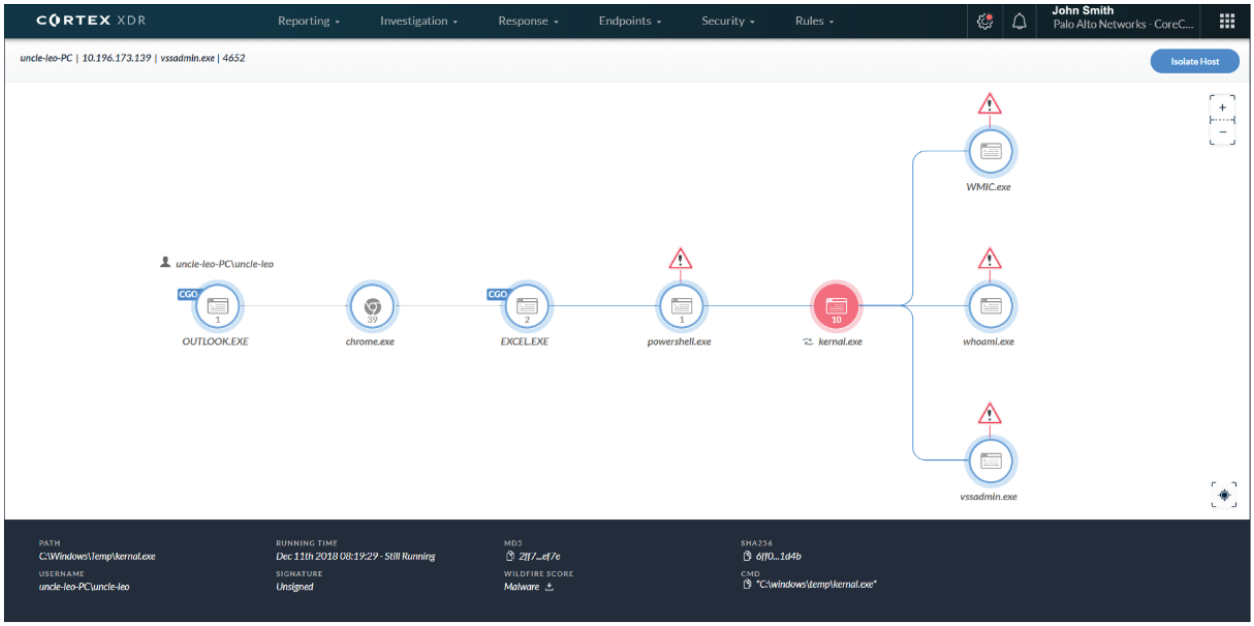
Native integration with network, endpoint, and cloud apps as well as WildFire threat intelligence



Cortex XDR reduces the time and experience required at every stage of security operations, from triage to threat hunting. Through tight integration with enforcement points like the Cortex XDR agent, it detects and contains threats quickly, and applies the knowledge gained to continually improve your security (see Figure 4-11).

Figure 4-11

Cortex XDR speeds alert triage and incident response.



Coordinated enforcement

The integrated suite of Palo Alto Networks products delivers greater security value than isolated components. Whenever a next-generation firewall sees a new piece of malware, or whenever an endpoint sees a new threat, protections are made available in minutes to all other next-generation firewalls and endpoints running the Cortex XDR agent, requiring no administrative effort whether it happens at 1:00 a.m. or 3:00 p.m. Tight integration between your network, endpoints, and clouds enables a continually improving security posture and provides coordinated enforcement to protect you from zero-day attacks.

Centralized logging across the platform

To surface evasive threats and prevent attacks, your organization must be able to perform advanced analytics as well as detection and response on all available data. Security applications that perform such analytics need access to scalable storage capacity and processing power.

Cortex Data Lake is a cloud-based storage offering for the context-rich, enhanced network and endpoint logs that Palo Alto Networks security products generate, including next-generation firewalls, Prisma Access, and the Cortex XDR agent. The cloud-based nature of Cortex Data Lake allows you to collect ever-expanding volumes of data without needing to plan for local compute and storage.

Cortex XDR uses Cortex Data Lake to store all event and incident data it captures, ensuring a clean handoff to other Palo Alto Networks products and services, such as Cortex XSOAR TIM contextual threat intelligence, for further investigation and incident response with endpoint context.

4.4.1.7 Cortex XDR technical architecture

The architecture of Cortex XDR is optimized for maximum availability, flexibility, and scalability to manage millions of endpoints. It comprises the following components:

- **Cortex XDR endpoint agent.** The endpoint agent consists of various drivers and services, but it requires only minimal memory and CPU usage—512MB of RAM and 200MB of disk space—to ensure a non-disruptive user experience. After it is deployed on your endpoints, your administrators have complete control over all Cortex XDR agents in your environment through the Cortex XDR console.

- **Cortex XDR management console.** Cortex XDR is a cloud-based application designed to minimize the operational challenges associated with protecting your endpoints. From the web-based Cortex XDR console, you can manage endpoint security policy, review security events as they occur, identify threat information, and perform additional analysis of associated logs.

- **WildFire malware prevention service.** Cortex XDR can send unknown malware to WildFire. Based on the properties, behaviors, and activities that a sample displays during analysis and execution in the WildFire sandbox, WildFire determines a verdict for the sample: benign, grayware, or malicious. WildFire then generates signatures and makes these globally available every five minutes, allowing other Palo Alto Networks products to recognize the newly discovered malware.

- **Cortex Data Lake.** Cortex Data Lake is a scalable, cloud-based log repository that stores context-rich logs generated by Palo Alto Networks security products, including next-generation firewalls, Prisma Access, and Cortex XDR agents. The cloud-based nature of Cortex Data Lake allows you to collect ever-expanding volumes of data without needing to plan for local compute and storage.

- **On-premises broker for restricted networks.** The on-premises broker service extends Cortex XDR agents to devices that cannot directly connect to the internet. Agents can use the broker service as a communication proxy to the Cortex XDR management service, receive the latest security console, and send content to Cortex Data Lake and WildFire without having to directly access the internet.

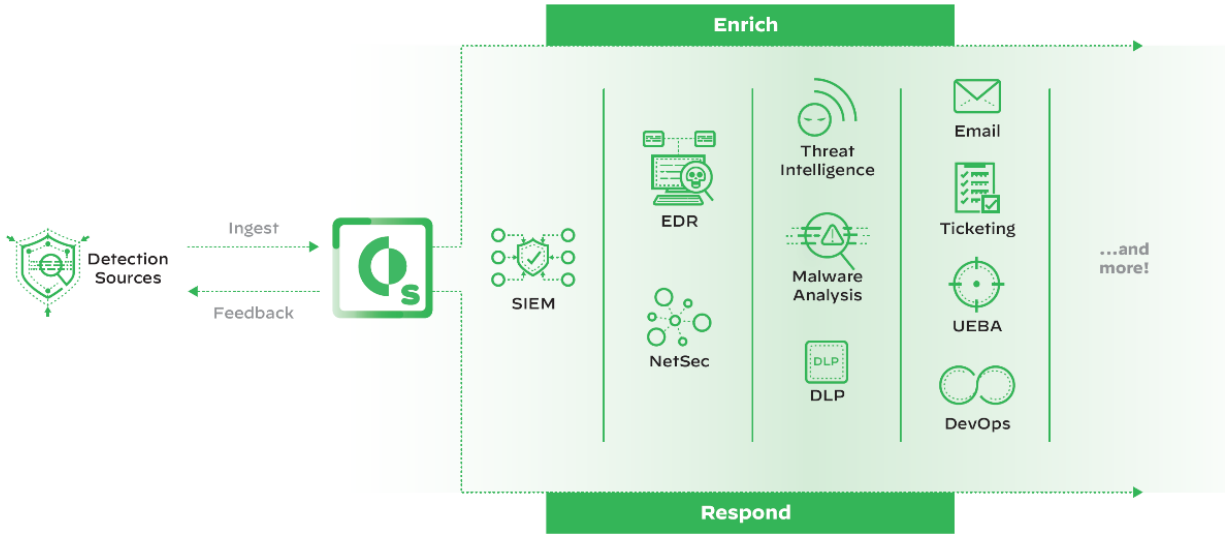
4.4.2 Cortex XSOAR

Security teams lack the people and scalable processes to keep pace with an overwhelming volume of alerts and endless security tasks. Analysts waste time pivoting across consoles for data collection, determining false positives, and performing manual, repetitive tasks throughout the lifecycle of an incident. As they face a growing skills shortage, security leaders need time to make decisions that matter, rather than drowning in reactive, piecemeal responses.

Cortex XSOAR supercharges security operations center (SOC) efficiency with the world's most comprehensive operating platform for enterprise security. Cortex XSOAR unifies case management, automation, real-time collaboration, and native threat intel management in the industry's first extended security orchestration, automation, and response (SOAR) offering. Teams can manage alerts across all sources, standardize processes with playbooks, take action on threat intelligence, and automate response for any security use case, resulting in up to 90 percent faster response times and as much as a 95 percent reduction in alerts requiring human intervention.

Cortex XSOAR ingests aggregated alerts and indicators of compromise (IoCs) from detection sources—such as security information and event management (SIEM) solutions, network security tools, threat intelligence feeds, and mailboxes—before executing automatable, process-driven playbooks to enrich and respond to these incidents (see Figure 4-12). These playbooks coordinate across technologies, security teams, and external users for centralized data visibility and action.

Figure 4-12



Cortex XSOAR ingests alerts and IoCs from multiple detection sources and executes playbooks to enrich and respond to incidents.

Cortex XSOAR empowers security professionals to efficiently carry out security operations and incident responses by streamlining security processes, connecting disparate security tools, and maintaining the right balance of machine-powered security automation and human intervention.

4.4.3 Threat intelligence (Cortex XSOAR TIM)

Highly automated and increasingly sophisticated cyberattacks are occurring in greater volume than ever before. Overburdened security teams, futilely attempting to investigate every threat in the enterprise network, have little time to analyze and understand truly advanced attacks.

Threat intelligence is at the core of every security operation. It applies to every security use case. Unfortunately, security teams are too overtaxed to truly take advantage of their threat intelligence, with thousands of alerts and millions of indicators coming at them daily. They require additional context, collaboration, and automation to extract true value. They need a solution that gives them the confidence to do their jobs effectively and shore up their defenses against the attacker's next move.

Cortex XSOAR Threat Intelligence Management (TIM) takes a unique approach to native threat intelligence management, unifying aggregation, scoring, and sharing of threat intelligence with playbook-driven automation.

Key features and capabilities in Cortex XSOAR TIM include:

- **Powerful, native centralized threat intel:** Supercharge investigations with instant access to the massive repository of built-in, high-fidelity Palo Alto Networks threat intelligence crowdsourced from the largest footprint of network, endpoint, and cloud intel sources—tens of millions of malware samples collected and firewall sessions analyzed daily.
- **Indicator relationships:** Indicator connections enable structured relationships to be created between threat intelligence sources and incidents. These relationships surface important context for security analysts on new threat actors and attack techniques.
- **Hands-free automated playbooks with extensible integrations:** Take automated action to shut down threats across more than 600 third-party products with purpose-built playbooks based on proven SOAR capabilities.

- **Granular indicator scoring and management:** Take charge of your threat intel with playbook-based indicator lifecycle management and transparent scoring that can be easily extended and customized.
- **Automated, multisource feed aggregation:** Eliminate manual tasks with automated playbooks to aggregate, parse, prioritize, and distribute relevant indicators in real time to security controls for continuous protection.
- **Most comprehensive marketplace:** The largest community of integrations with content packs that are prebuilt bundles of integrations, playbooks, dashboards, field subscription services, and all the dependencies needed to support specific security orchestration use cases. With 550+ integrations and 500+ product integrations, you can buy intel on the go using Marketplace points.

Cortex XSOAR TIM provides a common platform for incidents and threat information, where there is no disconnect between external threat data and your environment. Automated data enrichment of indicators provides analysts with relevant threat data to make smarter decisions.

4.4.4 Cortex Data Lake

Organizations often lack the visibility they need to stop attacks. Data is typically locked in silos across cloud, endpoint, and network assets, preventing tools from effectively finding, investigating, or automating threat response.

Deploying massive data collection, storage, and analysis infrastructure is complex. You need to plan for space, power, compute, networking, and high availability needs; increasing costs; and operational burden. After it is deployed, the infrastructure needs ongoing maintenance and monitoring, which takes time away from activities that drive your business forward.

Cortex Data Lake is built to benefit from the scale and locations of the public cloud. The cloud-based service is ready for elastic scaling from the start, eliminating the need for local compute and storage. As your needs grow, you can add more capacity with the push of a button. The public cloud architecture enables you to take advantage of global locations to solve local data residency and privacy requirements. Infrastructure—including storage and compute—is handled for you, letting you focus on solving new security challenges with apps built on Cortex.

Cortex Data Lake automatically collects, integrates, and normalizes data across your security infrastructure. With unified data, you can run advanced AI and machine learning to radically simplify security operations with apps built on Cortex. Tight sensor integration allows new data sources and types to be continually added to evolve your defenses.

Cortex Data Lake has strict privacy and security controls in place to prevent unauthorized access to sensitive or identifiable information. Cortex Data Lake ensures the privacy of your data by limiting access to your authorized users and apps, which you can revoke at any time. The Cortex Data Lake infrastructure is secured with industry-standard best practices for security and confidentiality, including rigorous technical and organizational security controls.

4.4.5 Cortex XSIAM

A major challenge in cybersecurity today is our inability to leverage massive scales of data for our defense. SIEM solutions have served security operations for many years as a way to aggregate and analyze alerts and logs—with incremental improvement in security outcomes. As compute and data storage have improved exponentially over the last decade, it is essential to radically reimagine how we can deliver real-time security that can match pervasive, AI-powered cyberattacks.

Built from the ground up as an autonomous security platform, Cortex XSIAM is an extended security intelligence and automation management platform that turns widespread infrastructure telemetry, threat intelligence, and external attack surface data into an intelligent data foundation to fuel best-in-class artificial intelligence and dramatically accelerate threat response. Unlike SIEM, Cortex XSIAM ingests granular data—not just alerts and logs—to fuel many layers of machine learning that automate critical threat detection and remediation steps downstream. And like any well-architected AI solution, the overall platform keeps getting better, learning from experience and outcomes.

4.4 Knowledge Check

Test your understanding of the fundamentals in the preceding section. Review the correct answers in Appendix A.

1. **True or False.** The key to Cortex XDR is blocking core exploit and malware techniques, not individual attacks.

2. **Fill in the Blank.** Cortex XDR Pro leverages _____ to analyze network, endpoint, and cloud data, which helps security analysts rapidly confirm threats by reviewing actionable alerts.

Appendix A – Knowledge Check Answers

Section 1.0 Knowledge Check

1. True
2. True
3. True
4. Health Insurance Portability and Accountability Act (HIPAA)
5. Discussion

Section 1.1 Knowledge Check

1. False. External threat actors have accounted for the majority of data breaches over the past five years.
2. Discussion
3. False. The cyberattack lifecycle is a seven-step process.
4. Reconnaissance, Weaponization, Delivery, Exploitation, Installation, Command and Control, Actions on the Objective
5. False. A defender needs to break only a single step in the cyberattack lifecycle framework to prevent an attack from succeeding.
6. [c] vulnerability and patch management
7. True

Section 1.2 Knowledge Check

1. [b] malware
2. [d] all of the above

Section 1.3 Knowledge Check

1. The primary issue with a perimeter-centric network security strategy is that it relies on the assumption that everything on the internal network can be trusted.

2. [b] least privilege

Section 1.4 Knowledge Check

1. Security Operating Platform
2. [a] network security, [b] advanced endpoint protection, [c] cloud security

Section 2.0 Knowledge Check

1. router
2. [b] Routing Information Protocol (RIP)
3. [a] distance-vector, [b] path-vector, and [c] link-state
4. True
5. Domain Name System (DNS)

Section 2.1 Knowledge Check

1. [a] IP address
2. 8
3. Subnetting

Section 2.2 Knowledge Check

1. [c] seven
2. [a] Transmission Control Protocol (TCP), [c] User Datagram Protocol (UDP)
3. media access control (MAC), Logical Link Control (LLC)
4. [a] Application, [b] Transport, [d] Internet, [e] Network Access
5. data encapsulation

Section 2.3 Knowledge Check

1. False. A dynamic packet filtering (also known as stateful packet inspection) firewall only inspects individual packet headers during session establishment to determine whether

the traffic should be allowed, blocked, or dropped by the firewall. After a session is established, individual packets that are part of the session are not inspected.

2. [a] proxies traffic rather than permitting direct communication between hosts, [b] can be used to implement strong user authentication, [c] masks the internal network from untrusted networks

3. [c] Secure Sockets Layer (SSL)

4. [c] It fully integrates all the security functions installed on the device.

Section 2.4 Knowledge Check

1. False. Signature-based anti-malware software is considered a reactive countermeasure because a signature file for new malware can't be created and delivered until the malware is already "in the wild."

2. Container-based

3. The main disadvantage of application allow listing related to exploit prevention is that an application that has been allow listed is permitted to run – even if the application has a vulnerability that can be exploited.

4. [a] data loss prevention (DLP), [b] policy enforcement, [d] malware prevention

Section 2.5 Knowledge Check

1. Identity and access management (IAM)

2. Open Systems Interconnection (OSI) model

Section 2.6 Knowledge Check

1. [b] adherence to strict port and protocol enforcement for allow or block decisions

2. application identification, user identification, and content identification

3. [a] packet headers

4. Any three of the following: security event log monitoring (Active Directory, Novell eDirectory, and Microsoft Exchange), user-provided credentials, client probing, receiving user information through XML API from an external LDAP directory

5. Unlike file-based malware scanning that waits until an entire file is loaded into memory to begin scanning, stream-based malware scanning begins scanning as soon as the first packets of the file are received. Stream-based malware scanning reduces latency and improves performance by receiving, scanning, and sending traffic to its intended destination immediately, without having to first buffer and then scan the file.

6. Templates eliminate manual, repetitive, risky, and error-prone configuration changes to multiple individual firewalls deployed throughout the enterprise network.

Section 3.0 Knowledge Check

1. [a] software as a service (SaaS)

2. hybrid

3. shared responsibility model

Section 3.1 Knowledge Check

1. hypervisor

2. [a] dormant VMs, [b] hypervisor vulnerabilities, [d] intra-VM communications

Section 3.2 Knowledge Check

1. Cloud, clusters, containers, code

2. [d] continuous deployment

3. Continuous integration

Section 3.3 Knowledge Check

1. Cloud computing doesn't mitigate existing network security risks; security requires isolation and segmentation, whereas the cloud relies on shared resources; security deployments are process-oriented, whereas cloud computing environments are dynamic.

2. [c] east-west

3. [b] consolidating servers within trust levels

Section 3.4 Knowledge Check

1. Prisma Cloud
2. The organizational security risks associated with unsanctioned SaaS application usage include regulatory noncompliance or compliance violations, loss of corporate intellectual property or other sensitive data, and malware distribution.
3. Traditional perimeter-based firewalls only have visibility of traffic that passes through the firewall. SaaS applications and data can be accessed from mobile devices that don't necessarily traverse a perimeter-based firewall, and many SaaS-based applications are designed to circumvent firewalls for performance and ease of use.
4. False. Prisma SaaS is used to protect sanctioned SaaS usage, as part of an integrated security solution that includes next-generation firewalls to prevent unsanctioned SaaS use. Prisma SaaS communicates directly with the SaaS applications themselves and therefore does not need to be deployed inline and does not require any software agents, proxies, additional hardware, or network configuration changes.
5. True

Section 4.0 Knowledge Check

1. Identify, investigate, mitigate
2. Interfaces

Section 4.1 Knowledge Check

1. Identify, investigate, mitigate, improve
2. [a] Identify
3. Tuning

Section 4.2 Knowledge Check

1. Security information and event management (SIEM) platform

Section 4.3 Knowledge Check

1. Security orchestration, automation, and response (SOAR)

Section 4.4 Knowledge Check

1. True
2. machine learning

Appendix B – Glossary

Address Resolution Protocol (ARP): A protocol that translates a logical address, such as an IP address, to a physical MAC address. RARP translates a physical MAC address to a logical address. See also *IP address*, *media access control (MAC) address*, and *Reverse Address Resolution Protocol (RARP)*.

Advanced Encryption Standard (AES): A symmetric block cipher based on the Rijndael cipher.

AES: See *Advanced Encryption Standard (AES).*

AI: See *artificial intelligence (AI).*

American Standard Code for Information Interchange (ASCII): A character-encoding scheme based on the English alphabet, consisting of 128 characters.

Android Package Kit (APK): An app created for the Android mobile operating system.

API: See *application programming interface (API).*

APK: See *Android Package Kit (APK).*

APP: See *Australian Privacy Principles (APP).*

application programming interface (API): A set of routines, protocols, and tools for building software applications and integrations.

AR: See *augmented reality (AR).*

ARP: See *Address Resolution Protocol (ARP).*

artificial intelligence (AI): The ability of a system or application to interact with and learn from its environment and automatically perform actions accordingly, without requiring explicit programming.

AS: See *autonomous system (AS).*

ASCII: See *American Standard Code for Information Interchange (ASCII).*

attack surface: Any area where breaches and exploits may occur. Comprised of an organization's entire digital footprint.

attack vector: A path or tool that an attacker uses to target a network. Also known as a threat vector.

augmented reality (AR): Augmented reality enhances a real-world environment with virtual objects.

Australian Privacy Principles (APP): The Privacy Act 1988 establishes standards for collecting and handling personal information, referred to as the Australian Privacy Principles (APP).

authoritative DNS server: The system of record for a given domain. See also *Domain Name System (DNS)*.

autonomous system (AS): A group of contiguous IP address ranges under the control of a single internet entity. Individual autonomous systems are assigned a 16-bit or 32-bit AS number (ASN) that uniquely identifies the network on the internet. ASNs are assigned by the Internet Assigned Numbers Authority (IANA). See also *Internet Protocol (IP) address* and *Internet Assigned Numbers Authority (IANA)*.

Bare-metal hypervisor: See *native hypervisor*.

BEC: See *business email compromise (BEC)*.

BES: See *bulk electric system (BES)*.

blockchain: A data structure containing transactional records (stored as blocks) that ensures security and transparency through a vast, decentralized peer-to-peer network with no single controlling authority. Cryptocurrency is an internet-based financial instrument that uses blockchain technology. See also *cryptocurrency*.

Boolean: A system of algebraic notation used to represent logical propositions.

boot sector: Contains machine code that is loaded into an endpoint's memory by firmware during the startup process, before the operating system is loaded.

boot sector virus: Targets the boot sector or master boot record (MBR) of an endpoint's storage drive or other removable storage media. See also *boot sector* and *master boot record (MBR)*.

bot: Individual endpoints that are infected with advanced malware that enables an attacker to take control of the compromised endpoint. Also known as a zombie. See also *botnet* and *malware*.

botnet: A network of bots (often tens of thousands or more) working together under the control of attackers using numerous command-and-control (C2) servers. See also *bot*.

bridge: A wired or wireless network device that extends a network or joins separate network segments.

bring your own access (BYOA): A remote-access policy in which remote users are allowed to connect to the corporate network using personal wireless service (for example, cellular service for a personal smartphone) from a wireless network operator.

bring your own device (BYOD): A policy trend in which organizations permit end users to use their own personal devices, primarily smartphones and tablets, for work-related purposes. BYOD relieves organizations of the cost of providing equipment to employees, but it creates a management challenge because of the vast number and type of devices that must be supported.

broadband cable: A type of high-speed internet access that delivers different upload and download data speeds over a shared network medium. The overall speed varies depending on the network traffic load from all the subscribers on the network segment.

broadcast domain: The portion of a network that receives broadcast packets sent from a node in the domain.

bulk electric system (BES): The large interconnected electrical system, consisting of generation and transmission facilities (among others), that comprises the "power grid."

bus topology: A LAN topology in which all nodes are connected to a single cable (the backbone) that is terminated on both ends. In the past, bus networks were commonly used for very small networks because they were inexpensive and relatively easy to install, but today bus topologies are rarely used. The cable media has physical limitations (the cable length), the backbone is a single point of failure (a break anywhere on the network affects the entire network), and tracing of a fault in a large network can be extremely difficult. See also *local-area network (LAN)*.

business email compromise (BEC): The unauthorized use of email leading to financial fraud. BEC techniques including spamming and phishing, among others.

BYOA: See *bring your own access (BYOA).*

BYOD: See *bring your own device (BYOD).*

California Consumer Privacy Act (CCPA): A privacy rights and consumer-protection statute for residents of California that was enacted in 2018 and became effective on January 1, 2020.

CASB: See *cloud access security broker (CASB).*

CCPA: See *California Consumer Privacy Act (CCPA).*

CD: See *continuous delivery (CD).*

CDN: See *content delivery network (CDN).*

child process: In multitasking operating systems, a subprocess created by a parent process that is currently running on the system.

CI: See *continuous integration (CI).*

CIDR: See *classless inter-domain routing (CIDR)*.

CIP: See *Critical Infrastructure Protection (CIP)*.

circuit-switched network: A network in which a dedicated physical circuit path is established, maintained, and terminated between the sender and receiver across a network for each communications session.

classless inter-domain routing (CIDR): A method for allocating IP addresses and IP routing that replaces classful IP addressing (for example, Class A, B, and C networks) with classless IP addressing. See also *Internet Protocol (IP) address*.

cloud access security broker (CASB): Software that monitors activity and enforces security policies on traffic between an organization's users and cloud-based applications and services.

collision domain: A network segment on which data packets may collide with each other during transmission.

consumerization: A computing trend that describes the process that occurs as end users increasingly find personal technology and apps that are more powerful or capable, more convenient, less expensive, quicker to install, and easier to use, than enterprise IT solutions.

container: A standardized, executable, and lightweight software code package that contains all the necessary components to run a given application (or applications) – including code, runtime, system tools and libraries, and configuration settings – in an isolated and virtualized environment to enable agility and portability of the application workload(s).

content delivery network (CDN): A network of distributed servers that distributes cached webpages and other static content to a user from a geographic location that is physically closest to the user.

continuous deployment: An automated CI pipeline that requires the code to pass automated testing before it is automatically deployed, giving customers instant access to new features. See also *continuous integration (CI)*.

continuous integration (CI): A development process that requires developers to integrate code into a repository several times per day for automated testing. Each check-in is verified by an automated build, allowing teams to detect problems early.

continuous delivery (CD): An automated CI pipeline that requires the code to go through manual technical checks before it is implemented into production. See also *continuous integration (CI)*.

convergence: The time required for all routers in a network to update their routing tables with the most current routing information about the network.

covered entity: Defined by HIPAA as a healthcare provider that electronically transmits PHI (such as doctors, clinics, psychologists, dentists, chiropractors, nursing homes, and pharmacies), a health plan (such as a health-insurance company, health maintenance organization, company health plan, or government program including Medicare, Medicaid, military and veterans' healthcare), or a healthcare clearinghouse. See also *Health Insurance Portability and Accountability Act (HIPAA)* and *protected health information (PHI)*.

CRC: See *cyclic redundancy check (CRC).*

Critical Infrastructure Protection (CIP): Cybersecurity standards defined by NERC to protect the physical and cyber assets necessary to operate the bulk electric system (BES). See also *bulk electric system (BES)* and *North American Electric Reliability Corporation (NERC)*.

cryptocurrency: A form of digital currency, such as Bitcoin, that uses encryption to control the creation of currency and verify the transfer of funds independent of a central bank or authority.

Cybersecurity Enhancement Act of 2014: A U.S. regulation that provides an ongoing, voluntary public-private partnership to improve cybersecurity and to strengthen cybersecurity research and development, workforce development and education, and public awareness and preparedness.

Cybersecurity Information Sharing Act (CISA): A U.S. regulation that enhances information sharing about cybersecurity threats by allowing internet traffic information to be shared between the U.S. government and technology and manufacturing companies.

cyclic redundancy check (CRC): A checksum used to create a message profile. The CRC is recalculated by the receiving device. If the recalculated CRC doesn't match the received CRC,

the packet is dropped, and a request to resend the packet is transmitted back to the device that sent the packet.

DAAS: Data, assets, applications, and services.

data encapsulation: A process in which protocol information from the OSI or TCP/IP layer immediately above is wrapped in the data section of the OSI or TCP/IP layer immediately below. Also referred to as data hiding. See also *Open Systems Interconnection (OSI) model* and *Transmission Control Protocol/Internet Protocol (TCP/IP) model*.

data hiding: See *data encapsulation.*

data mining: Enables patterns to be discovered in large datasets using machine learning, statistical analysis, and database technologies. See also *machine learning*.

DDoS: See *distributed denial of service (DDoS).*

default gateway: A network device, such as a router or switch, to which an endpoint sends network traffic when a specific destination IP address is not specified by an application or service, or when the endpoint does not know how to reach a specified destination. See also *router* and *switch*.

DevOps: The culture and practice of improved collaboration between application-development and IT operations teams.

DGA: See *domain generation algorithm (DGA).*

DHCP: See *Dynamic Host Configuration Protocol (DHCP).*

digital subscriber line (DSL): A type of high-speed internet access that delivers different upload and download data speeds. The overall speed depends on the distance from the home or business location to the provider's central office (CO).

distributed denial of service (DDoS): A type of cyberattack in which extremely high volumes of network traffic such as packets, data, or transactions are sent to the target victim's network to make their network and systems (such as an e-commerce website or other web application) unavailable or unusable.

DLL: See *dynamic-link library (DLL).*

DNS: See *Domain Name System (DNS).*

DNS over HTTPS (DoH): DNS traffic that is encrypted using the HTTPS protocol. See also *Domain Name System* (DNS) and *Hypertext Transfer Protocol Secure (HTTPS)*.

DoH: See *DNS over HTTPS (DOH)*.

domain generation algorithm (DGA): A program that is designed to generate domain names in a particular fashion. Attackers developed DGAs so that malware can quickly generate a list of domains that it can use for command and control (C2).

domain name registrar: An organization that is accredited by a TLD registry to manage domain name registrations. See also *top-level domain (TLD)*.

Domain Name System (DNS): A hierarchical distributed database that maps the FQDN for computers, services, or any resource connected to the internet or a private network to an IP address. See also *fully qualified domain name (FQDN)*.

drive-by download: A software download, typically malware, that happens without a user's knowledge or permission.

DSL: See *digital subscriber line (DSL).*

dylib hijacking: An exploit that injects malicious code into an application that searches for and loads dynamic libraries in a vulnerable manner. See also *dynamic-link library (DLL)*.

Dynamic Host Configuration Protocol (DHCP): A network-management protocol that dynamically assigns (leases) IP addresses and other network configuration parameters (such as default gateway and DNS information) to devices on a network. See also *default gateway* and *Domain Name System (DNS)*.

dynamic-link library (DLL): A type of file used in Microsoft operating systems that enables multiple programs to simultaneously share programming instructions contained in a single file to perform specific functions.

EAP: See *Extensible Authentication Protocol (EAP)*.

EAP-TLS: See *Extensible Authentication Protocol Transport Layer Security (EAP-TLS)*.

EBCDIC: See *Extended Binary-Coded Decimal Interchange Code (EBCDIC).*

EHR: See *electronic health record (EHR)*.

electronic health record (EHR): As defined by HealthIT.gov, an EHR "goes beyond the data collected in the provider's office and include[s] a more comprehensive patient history. EHR data can be created, managed, and consulted by authorized providers and staff from across more than one healthcare organization."

electronic medical record (EMR): As defined by HealthIT.gov, an EMR "contains the standard medical and clinical data gathered in one provider's office."

EMR: See *electronic medical record (EMR)*.

endpoint: A computing device such as a desktop or laptop computer, handheld scanner, IoT device or sensor (such as an autonomous vehicle, smart appliance, smart meter, smart TV, or wearable device), point-of-sale (POS) terminal, printer, satellite radio, security or videoconferencing camera, self-service kiosk, smartphone, tablet, or VoIP phone. Although endpoints can include servers and network equipment, the term is generally used to describe end-user devices. See also *internet of things (IoT)* and *Voice over Internet Protocol (VoIP)*.

Enterprise 2.0: A term introduced by Andrew McAfee and defined as "the use of emergent social software platforms within companies, or between companies and their partners or customers." See also *Web 2.0*.

exclusive or (XOR): A Boolean operator in which the output is true only when the inputs are different (for example, TRUE and TRUE equals FALSE, but TRUE and FALSE equals TRUE). See also *Boolean*.

exploit: A small piece of software code, part of a malformed data file, or a sequence (string) of commands that leverages a vulnerability in a system or software, causing unintended or unanticipated behavior in the system or software.

Extended Binary-Coded Decimal Interchange Code (EBCDIC): An 8-bit character-encoding scheme largely used on mainframe and midrange computers.

extended reality (XR): Broadly covers the spectrum from physical to virtual reality with various degrees of partial sensory to fully immersive experiences.

Extensible Authentication Protocol (EAP): A widely used authentication framework that includes about 40 different authentication methods.

Extensible Authentication Protocol Transport Layer Security (EAP-TLS): An Internet Engineering Task Force (IETF) open standard that uses the Transport Layer Security (TLS) protocol in Wi-Fi networks and PPP connections. See also *Internet Engineering Task Force (IETF)*, *Point-to-Point Protocol (PPP)*, and *Transport Layer Security (TLS)*.

Extensible Markup Language (XML): A programming-language specification that defines a set of rules for encoding documents in a human-readable and machine-readable format.

FaaS: See *function as a service (FaaS).*

false negative: In anti-malware, malware that is incorrectly identified as a legitimate file or application. In intrusion detection, a threat that is incorrectly identified as legitimate traffic. See also *false positive*.

false positive: In anti-malware, a legitimate file or application that is incorrectly identified as malware. In intrusion detection, legitimate traffic that is incorrectly identified as a threat. See also *false negative*.

favicon ("favorite icon"): A small file containing one or more small icons associated with a particular website or webpage.

Federal Exchange Data Breach Notification Act of 2015: A U.S. regulation that further strengthens HIPAA by requiring health-insurance exchanges to notify individuals whose personal information has been compromised as the result of a data breach as soon as possible, but no later than 60 days after breach discovery. See also *Health Insurance Portability and Accountability Act (HIPAA)*.

Federal Information Security Management Act (FISMA): See *Federal Information Security Modernization Act (FISMA)*.

Federal Information Security Modernization Act (FISMA): A U.S. law that implements a comprehensive framework to protect information systems used in U.S. federal government agencies. Known as the Federal Information Security Management Act prior to 2014.

fiber optic: Technology that converts electrical data signals to light and delivers constant data speeds in the upload and download directions over a dedicated fiber optic cable medium. Fiber optic technology is much faster and more secure than other types of network technology.

File Transfer Protocol (FTP): A program used to copy files from one system to another over a network.

FISMA: See *Federal Information Security Modernization Act (FISMA)*.

floppy disk: A removable magnetic storage medium commonly used from the mid-1970s until about 2007, when it was largely replaced by removable USB storage devices.

flow control: A technique used to monitor the flow of data between devices to ensure that a receiving device, which may not necessarily be operating at the same speed as the transmitting device, doesn't drop packets.

FQDN: See *fully qualified domain name (FQDN)*.

FTP: See *File Transfer Protocol (FTP).*

fully qualified domain name (FQDN): The complete domain name for a specific computer, service, or resource connected to the internet or a private network.

function as a service (FAAS): A cloud computing service that provides a platform for customers to develop, run, and manage their application functions without having to build and maintain the infrastructure normally required to develop and launch an application.

GDPR: See *General Data Protection Regulation (GDPR)*.

General Data Protection Regulation (GDPR): A European Union (EU) regulation that applies to any organization that does business with EU residents. It strengthens data protection for EU residents and addresses the export of personal data outside the EU.

Generic Routing Encapsulation (GRE): A tunneling protocol developed by Cisco Systems that can encapsulate various Network-layer protocols inside virtual point-to-point links.

GIF: See *Graphics Interchange Format (GIF).*

GLBA: See *Gramm-Leach-Bliley Act (GLBA)*.

Gramm-Leach-Bliley Act (GLBA): A U.S. law that requires financial institutions to implement privacy and information-security policies to safeguard the nonpublic personal information of clients and consumers.

Graphics Interchange Format (GIF): A bitmap image format that allows up to 256 colors and is suitable for images or logos (but not photographs).

GRE: See *Generic Routing Encapsulation (GRE)*.

hacker: Term originally used to refer to anyone with highly specialized computing skills, without connoting good or bad purposes. However, common misuse of the term has redefined a hacker as someone who circumvents computer security with malicious intent, such as a cybercriminal, cyberterrorist, or hacktivist.

hash signature: A cryptographic representation of an entire file or program's source code.

Health Insurance Portability and Accountability Act (HIPAA): A U.S. law that defines data privacy and security requirements to protect individuals' medical records and other personal health information. See also *covered entity* and *protected health information (PHI).*

heap spray: A technique used to facilitate arbitrary code execution by injecting a certain sequence of bytes into the memory of a target process.

hextet: A group of four 4-bit hexadecimal digits in a 128-bit IPv6 address. See also *Internet Protocol (IP) address*.

high-order bits: The first four bits in a 32-bit IPv4 address octet. See also *Internet Protocol (IP) address*, *octet*, and *low-order bits*.

HIPAA: See *Health Insurance Portability and Accountability Act (HIPAA).*

hop count: The number of router nodes that a packet must pass through to reach its destination.

hosted hypervisor: A hypervisor that runs within an operating-system environment. Also known as a Type 2 hypervisor. See also *hypervisor* and *native hypervisor*.

HTTP: See *Hypertext Transfer Protocol (HTTP).*

HTTPS: See *Hypertext Transfer Protocol Secure (HTTPS).*

hub: A device used to connect multiple networked devices together on a local-area network (LAN). Also known as a concentrator.

Hypertext Transfer Protocol (HTTP): An application protocol used to transfer data between web servers and web browsers.

Hypertext Transfer Protocol Secure (HTTPS): A secure version of HTTP that uses SSL or TLS encryption. See also *Secure Sockets Layer (SSL)* and *Transport Layer Security (TLS)*.

hypervisor: Technology that allows multiple, virtual (or guest) operating systems to run concurrently on a single physical host computer.

IaaS: See *Infrastructure as a service (IaaS)*.

IaC: See *infrastructure as code (IaC).*

IAM: See *identity and access management (IAM).*

IANA: See *Internet Assigned Numbers Authority (IANA).*

ICMP: See *Internet Control Message Protocol (ICMP)*.

IDE: See *integrated development environment (IDE).*

Identity and access management (IAM): A framework of business processes, policies, and technologies that facilitates the management of electronic or digital identities.

IETF: See *Internet Engineering Task Force (IETF)*.

IMAP: See *Internet Message Access Protocol (IMAP).*

indicator of compromise (IoC): A network or operating system (OS) artifact that provides a high level of confidence that a computer security incident has occurred.

infrastructure as a service (IaaS). A cloud-computing service model in which customers can provision processing, storage, networks, and other computing resources and deploy and run operating systems and applications. However, the customer has no knowledge of, and does not manage or control, the underlying cloud infrastructure. The customer has control over operating systems, storage, and deployed applications, along with some networking components (for example, host firewalls). The company owns the deployed applications and data, and it is therefore responsible for the security of those applications and data.

infrastructure as code (IaC): A DevOps process in which developers or IT operations teams can programmatically provision and manage the infrastructure stack (such as virtual machines, networks, and connectivity) for an application in software. See also *DevOps*.

initialization vector (IV): A random number used only once in a session, in conjunction with an encryption key, to protect data confidentiality. Also known as a nonce.

integrated development environment (IDE): A software application that provides comprehensive tools—such as a source-code editor, build automation tools, and a debugger—for application developers.

inter-process communication (IPC): A mechanism in an operating system that makes it possible to concurrently coordinate activities and manage shared data between different program processes.

Internet Assigned Numbers Authority (IANA): A private, nonprofit U.S. corporation that oversees global IP address allocation, autonomous system (AS) number allocation, root zone management in the Domain Name System (DNS), media types, and other Internet Protocol-related symbols and internet numbers. See also *autonomous system (AS)* and *Domain Name System (DNS)*.

Internet Control Message Protocol (ICMP): An internet protocol used to transmit diagnostic messages.

Internet Engineering Task Force (IETF): An open international community of network designers, operators, vendors, and researchers concerned with the evolution of the internet architecture and the smooth operation of the internet.

Internet Message Access Protocol (IMAP): A store-and-forward email protocol that allows an email client to access, manage, and synchronize email on a remote server.

internet of things (IoT): The IoT refers to the network of physical smart, connected objects that are embedded with electronics, software, sensors, and network connectivity.

Internet Protocol (IP) address: A 32-bit or 128-bit identifier assigned to a networked device for communications at the Network layer of the OSI model or the Internet layer of the TCP/IP model. See also *Open Systems Interconnection (OSI) model* and *Transmission Control Protocol/Internet Protocol (TCP/IP) model*.

intranet: A private network that provides information and resources – such as a company directory, human-resources policies and forms, department or team files, and other internal information – to an organization's users. Like the internet, an intranet uses the HTTP and/or HTTPS protocols, but access to an intranet is typically restricted to an organization's internal users. Microsoft SharePoint is a popular example of intranet software. See also *Hypertext Transfer Protocol (HTTP)* and *Hypertext Transfer Protocol Secure (HTTPS)*.

IoC: See *indicator of compromise (IoC)*.

IoT: See *internet of things (IoT)*.

IP address: See *Internet Protocol (IP) address*.

IP telephony: See *Voice over Internet Protocol (VoIP).*

IPC: See *inter-process communication (IPC).*

IV: See *initialization vector (IV)*.

jailbreaking: Hacking an Apple iOS device to gain root-level access to the device. This hacking is sometimes done by end users to allow them to download and install mobile apps without paying for them, from sources other than the App Store that are not sanctioned and/or controlled by Apple. Jailbreaking bypasses the security features of the device by replacing the firmware's operating system with a similar, albeit counterfeit version, which makes the device vulnerable to malware and exploits. See also *rooting*.

Joint Photographic Experts Group (JPEG): A photographic compression method used to store and transmit photographs.

JPEG: See *Joint Photographic Experts Group (JPEG).*

Just-in-Time (JIT) exploit: An exploit that targets a client-side JIT compiler used in Java programming.

Kerberos: An authentication protocol in which tickets are used to identify network users.

LAN: See *local-area network (LAN).*

least privilege: A network security principle in which only the permission or access rights necessary to perform an authorized task are granted.

least significant bit: The last bit in a 32-bit IPv4 address octet. See also *Internet Protocol (IP) address*, *octet*, and *most significant bit*.

linear bus topology: See *bus topology.*

LLC: See *Logical Link Control (LLC).*

local-area network (LAN): A computer network that connects laptop and desktop computers, servers, printers, and other devices so that applications, databases, files and file storage, and other networked resources can be shared across a relatively small geographic area such as a floor, a building, or a group of buildings.

Logical Link Control (LLC): A sublayer of the OSI model Data Link layer that manages the control, sequencing, and acknowledgement of frames and manages timing and flow control. See also *Open Systems Interconnection (OSI) model* and *flow control*.

Long-Term Evolution (LTE): A type of 4G cellular connection that provides fast connectivity primarily for mobile internet use.

low-order bits: The last four bits in a 32-bit IPv4 address octet. See also *Internet Protocol (IP) address*, *octet*, and *high-order bits*.

LTE: See *Long-Term Evolution (LTE).*

M2M: See *machine to machine (M2M).*

MAC address: See *media access control (MAC) address.*

machine learning: A subset of AI that applies algorithms to large datasets to discover common patterns in the data that can then be used to improve the performance of the system. See also *artificial intelligence (AI)*.

machine to machine (M2M): M2M devices are networked devices that exchange data and can perform actions without manual human interaction.

malware: Malicious software or code that typically damages, takes control of, or collects information from an infected endpoint. Malware broadly includes viruses, worms, Trojan horses (including Remote Access Trojans, or RATs), anti-AV, logic bombs, backdoors, rootkits, bootkits, spyware, and (to a lesser extent) adware.

master boot record (MBR): The first sector on a computer hard drive, containing information about how the logical partitions (or file systems) are organized on the storage media and an executable boot loader that starts up the installed operating system.

MBR: See *master boot record (MBR).*

MEC: See *multi-access edge computing (MEC).*

media access control (MAC) address: A unique 48-bit or 64-bit identifier assigned to a network interface card (NIC) for communications at the Data Link layer of the OSI model. See also *Open Systems Interconnection (OSI) model*.

metamorphism: A programming technique used to alter malware code with every iteration, to avoid detection by signature-based anti-malware software. Although the malware payload changes with each iteration – for example, by using a different code structure or sequence, or inserting garbage code to change the file size – the fundamental behavior of the malware payload remains unchanged. Metamorphism uses more advanced techniques than polymorphism. See also *polymorphism*.

MFA: See *multi-factor authentication* (MFA).

Microsoft Challenge-Handshake Authentication Protocol (MS-CHAP): A protocol used to authenticate Microsoft Windows-based workstations using a challenge-response mechanism to authenticate PPTP connections without sending passwords. See also *Point-to-Point Tunneling Protocol (PPTP)*.

mixed reality (MR): Includes technologies such as VR, AR, and XR that deliver an immersive and interactive physical and digital sensory experience in real time. See also *augmented reality (AR)*, *extended reality (XR)*, and *virtual reality (VR)*.

most significant bit: The first bit in a 32-bit IPv4 address octet. See also *Internet Protocol (IP) address*, *octet*, and *least significant bit*.

Motion Picture Experts Group (MPEG): An audio and video compression method used to store and transmit audio and video files.

MPEG: See *Motion Picture Experts Group (MPEG).*

MPLS: See *multiprotocol label switching (MPLS).*

MR: See *mixed reality (MR)*.

MS-CHAP: See *Microsoft Challenge-Handshake Authentication Protocol (MS-CHAP)*.

multi-access edge computing (MEC): MEC is defined by the European Telecommunications Standards Institute (ETSI) as an environment "characterized by ultra-low latency and high bandwidth as well as real-time access to radio network information that can be leveraged by applications."

multicloud: An enterprise cloud environment (or strategy) consisting of two or more public and/or private clouds.

multi-factor authentication (MFA): Any authentication mechanism that requires two or more of the following factors: something you know, something you have, something you are.

multiprotocol label switching (MPLS): MPLS is a networking technology that routes traffic using the shortest path based on "labels," rather than network addresses, to handle forwarding over private wide-area networks.

mutex: A program object that allows multiple program threads to share the same resource, such as file access, but not simultaneously.

NAT: See *network address translation (NAT).*

National Cybersecurity Protection Advancement Act of 2015: A U.S. regulation that amends the Homeland Security Act of 2002 to enhance multidirectional sharing of information related to cybersecurity risks and strengthens privacy and civil-liberties protections.

native hypervisor: A hypervisor that runs directly on the host computer hardware. Also known as a Type 1 or bare-metal hypervisor. See also *hypervisor* and *hosted hypervisor*.

natural language search: The ability to understand human spoken language and context, rather than using a Boolean search, for example, to find information. See also *Boolean*.

NERC: See *North American Electric Reliability Corporation (NERC)*.

network address translation (NAT): A technique used to virtualize IP addresses by mapping private, nonroutable IP addresses assigned to internal network devices to public IP addresses.

Network and Information Security (NIS) Directive: A European Union (EU) directive that imposes network and information security requirements for banks, energy companies, healthcare providers, and digital service providers, among others.

NIS Directive: See *Network and Information Security (NIS) Directive*.

nonce: See *initialization vector (IV).*

North American Electric Reliability Corporation (NERC): A not-for-profit international regulatory authority responsible for assuring the reliability of the bulk electric system (BES) in the continental United States, Canada, and the northern portion of Baja California, Mexico. See also *bulk electric system (BES)* and *Critical Infrastructure Protection (CIP)*.

obfuscation: A programming technique used to render code unreadable. It can be implemented using a simple substitution cipher, such as an XOR operation, or more sophisticated encryption algorithms, such as AES. See also *Advanced Encryption Standard (AES)*, *exclusive or (XOR)*, and *packer*.

octet: A group of eight bits in a 32-bit IPv4 address. See *Internet Protocol (IP) address*.

one-way hash function: A mathematical function that creates a unique representation (a hash value) of a larger set of data in a manner that is easy to compute in one direction (input to output), but not in the reverse direction (output to input). The hash function can't recover the original text from the hash value. However, an attacker could attempt to guess what the original text was and see if it produces a matching hash value.

Open Systems Interconnection (OSI) model: A seven-layer networking model consisting of the Application (Layer 7 or L7), Presentation (Layer 6 or L6), Session (Layer 5 or L5), Transport (Layer 4 or L4), Network (Layer 3 or L3), Data Link (Layer 2 or L2), and Physical (Layer 1 or L1) layers. Defines standard protocols for communication and interoperability using a layered approach in which data is passed from the highest layer (application) downward through each layer to the lowest layer (physical), then transmitted across the network to its destination, then passed upward from the lowest layer to the highest layer. See also *data encapsulation*.

optical carrier: A standard specification for the transmission bandwidth of digital signals on SONET fiber optic networks. Optical-carrier transmission rates are designated by the integer value of the multiple of the base rate (51.84Mbps). For example, OC-3 designates a 155.52Mbps (3 x 51.84) network, and OC-192 designates a 9953.28Mbps (192 x 51.84) network. See also *synchronous optical networking (SONET)*.

OSI model: See *Open Systems Interconnection (OSI) model.*

PaaS: See *platform as a service (PaaS)*.

packer: A software tool that can be used to obfuscate code by compressing a malware program for delivery, then decompressing it in memory at runtime. See also *obfuscation*.

packet capture (pcap): A traffic intercept of data packets that can be used for analysis.

packet-switched network: A network in which devices share bandwidth on communications links to transport packets between a sender and receiver across a network.

PAP: See *Password Authentication Protocol (PAP)*.

Password Authentication Protocol (PAP): An authentication protocol used by PPP to validate users with an unencrypted password. See also *Point-to-Point Protocol (PPP)*.

Payment Card Industry Data Security Standards (PCI DSS): A proprietary information security standard mandated and administered by the PCI Security Standards Council (SSC) and applicable to any organization that transmits, processes, or stores payment-card (such as debit and credit cards) information. See also *PCI Security Standards Council (SSC)*.

pcap: See *packet capture (pcap).*

PCI: See *Payment Card Industry Data Security Standards (PCI DSS).*

PCI DSS: See *Payment Card Industry Data Security Standards (PCI DSS).*

PCI Security Standards Council (SSC): A group comprising Visa, MasterCard, American Express, Discover, and JCB that maintains, evolves, and promotes PCI DSS. See also *Payment Card Industry Data Security Standards (PCI DSS)*.

PDU: See *protocol data unit (PDU).*

Personal Information Protection and Electronic Documents Act (PIPEDA): A Canadian privacy law that defines individual rights with respect to the privacy of their personal information and governs how private-sector organizations collect, use, and disclose personal information in the course of business.

personally identifiable information (PII): Defined by the U.S. National Institute of Standards and Technology (NIST) as "any information about an individual maintained by an agency, including (1) any information that can be used to distinguish or trace an individual's identity… and (2) any other information that is linked or linkable to an individual…."

pharming: A type of attack that redirects a legitimate website's traffic to a fake site.

PHI: See *protected health information (PHI).*

PII: See *personally identifiable information (PII).*

PIPEDA: See *Personal Information Protection and Electronic Documents Act (PIPEDA).*

PKI: See *public key infrastructure (PKI)*.

platform as a service (PaaS): A cloud-computing service model in which customers can deploy supported applications onto the provider's cloud infrastructure, but the customer has no knowledge of, and does not manage or control, the underlying cloud infrastructure. The customer has control over the deployed applications and limited configuration settings for the application-hosting environment. The company owns the deployed applications and data, and it is therefore responsible for the security of those applications and data.

playbooks: Task-based graphic workflows that help visualize processes across security products. Playbooks can be fully automated, fully manual, or anywhere in between. Also known as runbooks.

PoE: See *Power over Ethernet (PoE).*

Point-to-Point Protocol (PPP): A Layer 2 (Data Link) protocol layer used to establish a direct connection between two nodes.

Point-to-Point Tunneling Protocol (PPTP): An obsolete method for implementing virtual private networks, with many known security issues, that uses a TCP control channel and a GRE tunnel to encapsulate PPP packets. See also *Transmission Control Protocol (TCP)*, *Generic Routing Encapsulation (GRE)*, and *Point-to-Point Protocol (PPP)*.

polymorphism: A programming technique used to alter a part of malware code with every iteration to avoid detection by signature-based anti-malware software. For example, an encryption key or decryption routine may change with every iteration, but the malware payload remains unchanged. See also *metamorphism*.

POP3: See *Post Office Protocol Version 3 (POP3).*

Post Office Protocol Version 3 (POP3): An email-retrieval protocol that allows an email client to access email on a remote email server.

Power over Ethernet (PoE): A network standard that provides electrical power to certain network devices over Ethernet cables.

PPP: See *Point-to-Point Protocol (PPP)*.

PPTP: See *Point-to-Point Tunneling Protocol (PPTP).*

pre-shared key (PSK): A shared secret, used in symmetric key cryptography that has been exchanged between two parties communicating over an encrypted channel.

private cloud: A cloud-computing model that consists of a cloud infrastructure that is used exclusively by a single organization.

product integrations (or apps): Mechanisms through which SOAR platforms communicate with other products. These integrations can be executed through REST APIs, webhooks, and other techniques. An integration can be unidirectional or bidirectional, with the latter allowing both products to execute cross-console actions. See also *security orchestration, automation, and response (SOAR)*, *representational state transfer (REST)*, and *application programming interface (API)*.

protect surface: In a Zero Trust architecture, the protect surface consists of the most critical and valuable data, assets, application, and services (DAAS) on a network.

protected health information (PHI): Defined by HIPAA as information about an individual's health status, provision of healthcare, or payment for healthcare that includes identifiers such as names, geographic identifiers (smaller than a state), dates, phone and fax numbers, email addresses, Social Security numbers, medical record numbers, or photographs. See also *Health Insurance Portability and Accountability Act (HIPAA)*.

protocol data unit (PDU): A self-contained unit of data (consisting of user data or control information and network addressing).

PSK: See *pre-shared key (PSK)*.

public cloud: A cloud-computing deployment model that consists of a cloud infrastructure that is open to use by the general public.

public key infrastructure (PKI): A set of roles, policies, and procedures needed to create, manage, distribute, use, store, and revoke digital certificates and manage public key encryption.

QoS: See *quality of service (QoS)*.

quality of service (QoS): The overall performance of specific applications or services on a network, including error rate, bit rate, throughput, transmission delay, availability, jitter, etc. QoS policies can be configured on certain network and security devices to prioritize certain traffic, such as voice or video, over other, less performance-intensive traffic, such as file transfers.

RADIUS: See *Remote Authentication Dial-In User Service (RADIUS)*.

rainbow table: A precomputed table used to find the original value of a cryptographic hash function.

RARP: See *Reverse Address Resolution Protocol (RARP).*

RASP: See *runtime application self-protection (RASP).*

RBAC: See *role-based access control (RBAC).*

recursive DNS query: A DNS query that is performed (if the DNS server allows recursive queries) when a DNS server is not authoritative for a destination domain. The nonauthoritative DNS server obtains the IP address of the authoritative DNS server for the destination domain and sends the original DNS request to that server to be resolved. See also *Domain Name System (DNS)* and *authoritative DNS server*.

Remote Authentication Dial-In User Service (RADIUS): A client-server protocol and software that enables remote-access servers to communicate with a central server to authenticate users and authorize access to a system or service.

Remote Procedure Call (RPC): An inter-process communication (IPC) protocol that enables an application to be run on a different computer or network, rather than on the local computer on which it is installed. See also *inter-process communication (IPC)*.

repeater: A network device that boosts or retransmits a signal to physically extend the range of a wired or wireless network.

representational state transfer (REST): An architectural programming style that typically runs over HTTP and is commonly used for mobile apps, social-networking websites, and mashup tools. See also *Hypertext Transfer Protocol (HTTP)*.

REST: See *representational state transfer (REST)*.

return-oriented programming (ROP) exploit: An exploit that allows an attacker to take control of a program by executing code that diverts the execution flow of the program.

Reverse Address Resolution Protocol (RARP): A protocol that translates a physical MAC address to a logical address. See also *media access control (MAC) address*.

ring topology: A LAN topology in which all nodes are connected in a closed loop that forms a continuous ring. In a ring topology, all communication travels in a single direction around the ring. Ring topologies were common in token ring networks. See also *local-area network (LAN)*.

role-based access control (RBAC): A method for implementing discretionary access controls in which access decisions are based on group membership, according to organizational or functional roles.

rooting: The Google Android equivalent of jailbreaking. See *jailbreaking*.

router: A network device that sends data packets to a destination network along a network path.

RPC: See *remote procedure call (RPC)***.**

runtime application self-protection (RASP): Technology that detects attacks against an application in real time. RASP continuously monitors an app's behavior and the context of behavior to immediately identify and prevent malicious activity.

SaaS: See *software as a service (SaaS).*

salt: Randomly generated data that is used as an additional input to a one-way hash function that hashes a password or passphrase. The same original text hashed with different salts results in different hash values. See also *one-way hash function*.

Sarbanes-Oxley (SOX) Act: A U.S. law that increases financial governance and accountability in publicly traded companies.

SASE: See *Secure Access Service Edge (SASE).*

SCM: See *software configuration management (SCM).*

script kiddie: Someone with limited hacking and/or programming skills who uses malicious programs (malware) written by others to attack a computer or network. See also *malware*.

SCTP: See *Stream Control Transmission Protocol (SCTP).*

SD-WAN: See *software-defined wide-area network (SD-WAN).*

Secure Access Service Edge (SASE): An integrated solution that provides consistent networking and security services and access to cloud applications delivered through a common framework.

Secure Shell (SSH): A more secure alternative to Telnet for remote access. SSH establishes an encrypted tunnel between the client and the server, and it can also authenticate the client to the server. See also *Telnet*.

Secure Sockets Layer (SSL): A cryptographic protocol for managing authentication and encrypted communication between a client and server to protect the confidentiality and integrity of data exchanged in the session.

secure web gateway (SWG): A security platform or service that is designed to maintain visibility in web traffic. Additional functionality may include web content filtering.

security orchestration, automation, and response (SOAR): Technology that helps coordinate, execute, and automate tasks between various people and tools, allowing companies to respond quickly to cybersecurity attacks and improve their overall security posture. SOAR tools use playbooks to automate and coordinate workflows that may include any number of disparate security tools as well as human tasks. See also *playbook*.

serverless: Generally refers to an operational model in cloud computing in which applications rely on managed services that abstract away the need to manage, patch, and secure infrastructure and virtual machines. Serverless applications rely on a combination of managed cloud services and FaaS offerings. See also *function as a service (FaaS)*.

service set identifier (SSID): A case-sensitive, 32-character alphanumeric identifier that uniquely identifies a Wi-Fi network.

Session Initiation Protocol (SIP): An open signaling protocol standard for establishing, managing, and terminating real-time communications—such as voice, video, and text—over large IP-based networks.

Simple Mail Transfer Protocol (SMTP): A protocol used to send and receive email across the internet.

Simple Network Management Protocol (SNMP): A protocol used to collect information by polling stations and sending traps (or alerts) to a management station.

SIP: See *Session Initiation Protocol (SIP).*

SMTP: See *Simple Mail Transfer Protocol (SMTP).*

SNMP: See *Simple Network Management Protocol (SNMP).*

SOAR: See *security orchestration, automation, and response (SOAR).*

software as a service (SaaS): A category of cloud-computing services in which the customer is provided access to a hosted application that is maintained by the service provider.

software-defined wide-area network (SD-WAN): A virtualized service that separates the network control and management processes from the underlying hardware in a wide-area network and makes those processes available as software.

software configuration management (SCM): The task of tracking and controlling changes in software.

SONET: See *synchronous optical networking (SONET).*

SOX: See *Sarbanes-Oxley (SOX) Act*.

spear phishing: A highly targeted phishing attack that uses specific information about the target to make the phishing attempt appear legitimate.

SSH: See *Secure Shell (SSH)*.

SSID: See *service set identifier (SSID)*.

SSL: See *Secure Sockets Layer (SSL).*

Stream Control Transmission Protocol (SCTP): A message-oriented protocol (similar to UDP) that ensures reliable, in-sequence transport with congestion control (similar to TCP). See also *User Datagram Protocol (UDP)* and *Transmission Control Protocol (TCP)*.

subnet mask: A number that hides the network portion of an IPv4 address, leaving only the host portion of the IP address. See also *Internet Protocol (IP) address*.

subnetting: A technique used to divide a large network into multiple smaller subnetworks.

supernetting: A technique used to aggregate multiple contiguous smaller networks into a larger network to enable more efficient internet routing.

SWG: See *secure web gateway (SWG).*

switch: An intelligent hub that forwards data packets only to the port associated with the destination device on a network.

synchronous optical networking (SONET): A protocol that transfers multiple digital bit streams synchronously over optical fiber.

T-carrier: A full-duplex digital transmission system that uses multiple pairs of copper wire to transmit electrical signals over a network. For example, a T-1 circuit consists of two pairs of copper wire—one pair transmits, the other pair receives—that are multiplexed to provide a total of 24 channels, each delivering 64Kbps of data, for a total bandwidth of 1.544Mbps.

TCP: See *Transmission Control Protocol (TCP).*

TCP segment: A PDU defined at the Transport layer of the OSI model. See also *protocol data unit (PDU)* and *Open Systems Interconnection (OSI) model*.

TCP/IP model: See *Transmission Control Protocol/Internet Protocol (TCP/IP) model*.

technical debt: A software-development concept, which has also been applied more generally to IT, in which additional future costs are anticipated for rework due to an earlier decision or

course of action that was necessary for agility, but not necessarily the most optimal or appropriate decision or course of action.

Telnet: A terminal emulator used to provide remote access to a system.

three-way handshake: A sequence used to establish a TCP connection. For example, a PC initiates a connection with a server by sending a TCP SYN (Synchronize) packet. The server replies with a SYN ACK packet (Synchronize Acknowledgment). Finally, the PC sends an ACK or SYN-ACK-ACK packet, acknowledging the server's acknowledgement, and data communication commences. See also *Transmission Control Protocol (TCP)*.

threat vector: See *attack vector.*

TLD: See *top-level domain (TLD).*

TLS: See *Transport Layer Security (TLS)*.

top-level domain (TLD): The highest-level domain in DNS, represented by the last part of an FQDN (for example, .com or .edu). The most commonly used TLDs are generic top-level domains (gTLD) such as .com, edu, .net, and .org, and country-code top-level domains (ccTLD) such as .ca and .us. See also *Domain Name System (DNS)*.

Transmission Control Protocol (TCP): A connection-oriented (a direct connection between network devices is established before data segments are transferred) protocol that provides reliable delivery (received segments are acknowledged, and retransmission of missing or corrupted segments is requested) of data.

Transmission Control Protocol/Internet Protocol (TCP/IP) model: A four-layer networking model consisting of the Application (Layer 4 or L4), Transport (Layer 3 or L3), Internet (Layer 2 or L2), and Network Access (Layer 1 or L1) layers.

Transport Layer Security (TLS): The successor to SSL (although it is still commonly referred to as SSL). See also *Secure Sockets Layer (SSL)*.

Type 1 hypervisor: See *native hypervisor*.

Type 2 hypervisor: See *hosted hypervisor.*

UDP: See *User Datagram Protocol (UDP).*

UDP datagram: A PDU defined at the Transport layer of the OSI model. See also *protocol data unit* (PDU), *User Datagram Protocol (UDP)*, and *Open Systems Interconnection (OSI) model*.

UEBA: See *user and entity behavior analytics (UEBA).*

user and entity behavior analytics (UEBA): A type of cybersecurity solution or feature that discovers threats by identifying activity that deviates from a normal baseline.

uniform resource identifier (URI): A string of characters that uniquely identifies a resource, using a predefined syntax in a hierarchical naming scheme.

uniform resource locator (URL): A unique reference (or address) to an internet resource, such as a webpage.

URI: See *uniform resource identifier (URI).*

URL: See *uniform resource locator (URL)*.

User Datagram Protocol (UDP): A connectionless (a direct connection between network devices is not established before datagrams are transferred) protocol that provides best-effort delivery (received datagrams are not acknowledged, and missing or corrupted datagrams are not requested) of data.

variable-length subnet masking (VLSM): A technique that enables IP address spaces to be divided into different sizes. See also *Internet Protocol (IP) address*.

virtual local-area network (VLAN): A logical network that is created within a physical local-area network.

virtual machine (VM): An emulation of a physical (hardware) computer system, including CPU, memory, disk, operating system, network interfaces, etc.

virtual reality (VR): A simulated digital experience.

VLAN: See *virtual local-area network (VLAN)*.

VLSM: See *variable-length subnet masking (VLSM)*.

VM: See *virtual machine (VM).*

Voice over Internet Protocol (VoIP): Technology that provides voice communication over an Internet Protocol (IP)-based network. Also known as IP telephony.

VoIP: See *Voice over Internet Protocol (VoIP)*.

VR: See *virtual reality (VR).*

vulnerability: A bug or flaw that exists in a system or software and creates a security risk.

WAN: See *wide-area network (WAN).*

watering hole: An attack that compromises websites that are likely to be visited by a targeted victim to deliver malware via a drive-by download. See also *drive-by download*.

Web 2.0: A term popularized by Tim O'Reilly and Dale Dougherty unofficially referring to a new era of the World Wide Web, which is characterized by dynamic or user-generated content, interaction, and collaboration, and the growth of social media. See also *Enterprise 2.0*.

Web 3.0: As defined on ExpertSystem.com, Web 3.0 is characterized by the following five characteristics: Semantic web, artificial intelligence, 3D graphics, connectivity, and ubiquity.

whaling: A type of spear-phishing attack that is specifically directed at senior executives or other high-profile targets within an organization. See also *spear phishing*.

wide-area network (WAN): A computer network that connects multiple LANs or other WANs across a relatively large geographic area, such as a small city, a region or country, a global enterprise network, or the entire planet (for example, the internet). See also *local-area network (LAN)*.

wireless repeater: A device that rebroadcasts the wireless signal from a wireless router or AP to extend the range of a Wi-Fi network.

XML: See *Extensible Markup Language (XML)*.

XOR: See *exclusive or (XOR)*.

XR: See *extended reality (XR).*

zero-day threat: The window of vulnerability that exists from the time a new (unknown) threat is released until security vendors release a signature file or security patch for the threat.

zombie: See *bot*.

Appendix C – Palo Alto Networks Technical Training and Certification Programs

Palo Alto Networks Technical Training Program

Palo Alto Networks offers a technical training program that provides you with the advanced knowledge you need to secure enterprise networks and safely enable applications. Training from Palo Alto Networks and Palo Alto Networks Authorized Training Centers delivers knowledge and expertise that prepare you to protect our digital way of life. You can learn more about this program at www.paloaltonetworks.com/services/education.

Palo Alto Networks Certification Program

Palo Alto Networks trusted security certifications validate your knowledge of the Palo Alto Networks Security Operating Platform and your ability to help prevent successful cyberattacks and safely enable applications. The three primary certifications in the program are outlined below, and you can learn more about this program at www.paloaltonetworks.com/services/education/certification.html.

Palo Alto Networks Certified Cybersecurity Entry-Level Technician (PCCET)

PCCET, the entry-level Palo Alto Networks certification, is designed for students, technical professionals, and non-technical professionals interested in validating comprehensive knowledge on current cybersecurity tenets. The PCCET is a knowledge-based certification on the fundamentals of cybersecurity that will stand as the entry point in accessing the entire Palo Alto Networks credentialing portfolio. The certification will assess the individual's knowledge of firewalls as well as the cloud and automation functionalities of Prisma and Cortex.

Palo Alto Networks Certified Network Security Associate (PCNSA)

PCNSA, the intermediate-level Palo Alto Networks certification, is for security administrators and professionals looking to validate their knowledge of Palo Alto Networks next-generation firewall and their ability to configure central features, using User-ID, App-ID, and Content-ID technology to inform policy. PCNSA-certified individuals stand out with validated expertise on using the industry's best firewalls to protect networks from cutting-edge cyberthreats. Employers of certified practitioners gain administrators and analysts who can reliably practice

advanced security methods and threat detection with next-generation firewalls to enable effective cybersecurity in any work setting.

Palo Alto Networks Certified Network Security Engineer (PCNSE)

PCNSE certification is the third and final step on the Palo Alto Networks certification pathway. This expert-level certification is intended for security engineers and cyber- security professionals skilled in designing, deploying, configuring, maintaining, and troubleshooting the majority of Security Operating Platform implementations.

PCNSE certification exhibits the highest level of expertise in the Security Operating Platform, validating the certified practitioner's skill in managing the industry's leading cybersecurity practices and working with world-renowned security products. PCNSE-certified experts are highly valued for their capabilities in implementing advanced cybersecurity in all types of operations.

Certified practitioners have proven their expert-level experience with Palo Alto Networks next-generation firewalls, including the ability to effectively engineer, deploy, configure, maintain, and troubleshoot the majority of Security Operating Platform implementation scenarios. By extension, PCNSE certification validates an expert's ability to use Palo Alto Networks technology to prevent successful cyberattacks across the digital world.

www.paloaltonetworks.com/academy

Advisory Panel:

Jim Boardman

Keith Cantillon

James Dalton

Abhir Naik

Saravanan Rajagopal

Thomas Trevethan

© 2022 Palo Alto Networks, Inc. – all rights reserved.

Palo Alto Networks is a registered trademark of Palo Alto Networks.

A list of our trademarks can be found at https://www.paloaltonetworks.com/company/trademarks.html

All other marks mentioned herein may be trademarks of their respective companies.