

Zero Trust in Resilient Cloud and Network Architectures



ciscopress.com

JOSH HALLEY
DHRUMIL PRAJAPATI
ARIEL LEZA
VINAY SAINI

FREE SAMPLE CHAPTER |



Zero Trust in Resilient Cloud and Network Architectures

Josh Halley, CCIEx3 No. 11924

Dhrumil Prajapati, CCIEx2 No. 28071,
CCDE No. 20210002

Ariel Leza

Vinay Saini, CCIE No. 38448,
CWNE No. 69, CCDE No. 20240032

Cisco Press

Zero Trust in Resilient Cloud and Network Architectures

Josh Halley, Dhrumil Prajapati, Ariel Leza, Vinay Saini

Copyright © 2025 Cisco Systems, Inc.

Published by:
Cisco Press
Hoboken, New Jersey

All rights reserved. This publication is protected by copyright, and permission must be obtained from the publisher prior to any prohibited reproduction, storage in a retrieval system, or transmission in any form or by any means, electronic, mechanical, photocopying, recording, or likewise. For information regarding permissions, request forms, and the appropriate contacts within the Pearson Education Global Rights & Permissions Department, please visit <https://www.pearson.com/global-permission-granting.html>.

No patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this book, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein.

Please contact us with concerns about any potential bias at www.pearson.com/en-us/report-bias.html.

\$PrintCode

Library of Congress Control Number: 2025934165

ISBN-13: 978-0-13-820460-0

ISBN-10: 0-13-820460-8

Warning and Disclaimer

This book is designed to provide information about segmentation concepts, network access control, and resilient cloud and enterprise network architectures. Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied.

The information is provided on an “as is” basis. The authors, Cisco Press, and Cisco Systems, Inc. shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the discs or programs that may accompany it.

The opinions expressed in this book belong to the authors and are not necessarily those of Cisco Systems, Inc.

Trademark Acknowledgments

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Cisco Press or Cisco Systems, Inc., cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Feedback Information

At Cisco Press, our goal is to create in-depth technical books of the highest quality and value. Each book is crafted with care and precision, undergoing rigorous development that involves the unique expertise of members from the professional technical community.

Readers' feedback is a natural continuation of this process. If you have any comments regarding how we could improve the quality of this book, or otherwise alter it to better suit your needs, you can contact us through email at feedback@ciscopress.com. Please make sure to include the book title and ISBN in your message.

We greatly appreciate your assistance.

Please contact us with concerns about any potential bias at <https://www.pearson.com/report-bias.html>.

Head of IT & Professional Learning, Enterprise Learning and Skills: Julie Phifer

Alliances Manager, Cisco Press: Caroline Antonio

Executive Editor: James Manly

Managing Editor: Sandra Schroeder

Development Editor: Ellie C. Bru

Senior Project Editor: Mandie Frank

Copy Editor: Chuck Hutchinson

Technical Editors: Asier Arlegui Lacunza, Istvan Matyasovszki

Designer: Chuti Prasertsith

Composition: codeMantra

Indexer: Timothy Wright

Proofreader: Donna E. Mulder



Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

About the Authors

Josh Halley is a Principal Architect in the office of the CTO at Cisco Systems, where his current role focuses on domains related to up-and-coming and burgeoning new technology trends and innovation, engaging in large and complex strategic negotiations, and working with C-Levels to set technology strategy and direction within their organizations. Over the years, Josh has worked in many differing roles at organizations ranging from technology companies to banking and finance and management consulting.

Technology has long been a passion and key area of interest throughout his career, leading Josh to pursue three CCIE certifications to advance his expertise across multiple domains. During his years at Cisco, he has been engaged in technology projects and pre-sales activities all around the world, which has given him the opportunity to learn different cultures, perspectives, and customs.

Within Cisco, Josh has worked side by side with many of the company's business units, product managers, and technical marketing engineers and has been directly involved in the creation and deployment of over 180 new software features across multiple technology domains, many of which are associated with zero trust capabilities seen in Cisco's products and portfolio today.

Today Josh maintains his focus on technology, driving innovation with customers in the field and actively participating in the creation of patents and white papers and new product and service offerings for Cisco. Further to his work within Cisco, Josh is also an open-source software advocate who actively participates in working groups from the Cloud Native Computing Foundation, providing him an opportunity to share his knowledge and expertise with the wider open-source community.

Dhrumil Prajapati is a Principal Architect within Cisco CX's GES Architectures team. His focus lies on designing multidomain networks, and he has been offering a complete lifecycle of professional services and architecture advice for the past 15 years. His expertise extends to serving enterprise, government, and service provider entities across the globe. His services are designed to assist clients in planning, designing, deploying, managing, and interoperating all networking technology domains within their private or public infrastructure and application environments.

In his networking career, Dhrumil has designed networks for over 175 organizations, which inspired him to write a book on the subject. He is a coauthor of *Designing Real-World Multi-Domain Networks* (<https://www.ciscopress.com/store/designing-real-world-multi-domain-networks-9780138037215>) and *Cisco SD-Access for Industry Verticals* (<https://cs.co/sda-verticals-book>). He also holds patents and has given multiple presentations in Cisco Live on SD-Access, multidomain, and automation.

Dhrumil holds dual CCIEs in Enterprise Infrastructure and Service Provider, as well as a CCDE, in addition to other leading technical certificates. He also assists the Cisco Certifications team by reviewing and providing feedback for Cisco Certification Program. In addition, he leads several initiatives within Cisco CX aimed at driving delivery standardization and enhancing efficiency through automation innovation.

Currently residing in Apex, North Carolina, Dhrumil has a passion for motor racing, woodworking, and innovative electronics that enhance human life. His wife, Devanshi, and son, Ram, bring pure joy to his life, adding a touch of fun every day.

Ariel Leza, an ardent entrepreneur and technology aficionado, currently excels as a Cloud Solutions Architect in the EMEA CX CTO office at Cisco. His expertise is centered on guiding clients toward achieving their unique objectives through intricate cloud and IT architectures. Ariel's journey in the IT realm commenced during his military service as a network engineer in 2011, uncovering his passion for networking and cybersecurity. Today, he stands as a results-driven entrepreneur with a proven track record in technical sales and product positioning, fueled by enthusiasm for creating new businesses.

With over a decade of experience in the field, Ariel specializes in cloud and cross-domain IT architectures, focusing on cloud-native technologies, open-source software, networking, virtualization/compute, and data center software stack components. As a technology leader and technical solution architect in the Israeli market, he has demonstrated an exceptional ability to convey complex technological concepts in simple terms, tailoring his message to diverse audiences and emphasizing Cisco's unique value proposition.

In his role under the Customer Experience CTO Office of EMEA, Ariel has been instrumental in driving strategic programs and building strategic relationships with key accounts, leading to significant achievements such as engagement with top EMEA accounts, patent submissions, and pioneering deployments in blockchain technology and cloud-native services. His skills in cloud-native platforms, DevOps, Linux, and cloud computing are complemented by his deep understanding of decentralized applications, cryptocurrency, microservices, and Web3 blockchain architectures.

Outside of his professional endeavors, Ariel is a passionate participant in open-source and community projects. He also dedicates his time to global business ventures and the management of his demo Innovation Lab initiative. Ariel's commitment to making a meaningful impact as an entrepreneur and businessperson is driven by his belief in taking active responsibility for our civilization's actions and consequences.

With a vibrant spirit and an undying passion for technology and innovation, Ariel Leza continues to make significant strides in the tech world, inspiring others and driving successful customer outcomes with his dynamic approach and creative solutions.

Vinay Saini is a technologist, inventor, author, and mentor with over two decades of experience in the networking industry. As a Principal Architect at Cisco Systems (Customer Experience Group), Vinay guides customers across diverse verticals—including enterprise and IIoT—on their digital transformation journeys.

Vinay holds a bachelor of technology degree in IT and an MBA in international business. He is the first individual in India to achieve dual expert-level certifications: CWNE (#69) and CCIE (#38448). Additionally, Vinay is CCDE certified (#20240032) and holds many other industry credentials. He also actively contributes to Cisco Certification programs.

With a portfolio of over 100 patents filed and numerous defensive innovations, Vinay is at the forefront of technological innovation. As a sought-after speaker at events like Cisco Live, Vinay is also passionate about mentoring professionals worldwide, fostering their growth in technical and behavioral domains to help them reach their full potential.

About the Technical Reviewers

Asier Arlegui Lacunza, CCIE No. 5921, has been with Cisco since 1998 and currently works as a Principal Architect in Cisco's Customer Experience organization. In the past 20+ years of his career at Cisco, he has worked as a technical architect on a wide range of enterprise (data center, campus, and enterprise WAN) and service provider (access and core networking) technology projects, with a focus on network automation. He holds a master's degree in telecommunications engineering from Public University of Navarre, Spain.

Istvan Matyasovszki has been with Cisco since 2007 and is currently a Security Solutions Architect in Cisco's EMEA Customer Experience organization, with a special interest and focus on network access control, remote access, intrusion detection, and microsegmentation. Istvan holds a PhD in computer engineering from the University of Limerick in Ireland and, before that, worked as a UNIX/Linux system administrator for several years.

Dedications

Josh:

I would like to dedicate this book to my two bloodhounds. Thanks for keeping me company and spending many a late night and early morning with me while I was authoring my chapters.

Dhrumil:

Publishing a first book is a core memory, and the third one sets an example for many. I want to dedicate this book to my wife, Devanshi, who has given unconditional love and been a true partner as I achieve my goal. It goes to the silent sacrifices you have made that brought me to where I am today.

My son, Ram, I never imagined I would write a book, but your curiosity in reading books from a young age and gaining as much knowledge as possible has inspired me to write my third book.

Ariel:

I would like to dedicate this book to the open-source community, including the developers and advocates championing the progression of the next generation of decentralized technologies.

Vinay:

To my parents, for instilling in me the values of perseverance and curiosity.

To my wife, Sowbhagya, for her unwavering support and patience throughout this journey.

To my children, Vihaan and Pranav, for inspiring me to think about a better future.

To my colleagues, friends, and mentors, whose insights, encouragement, and collaboration have been invaluable.

This book is dedicated to you.

Acknowledgments

Josh:

There were so many talented individuals who supported me on the journey of writing this publication that I feel it could fill an entire new chapter alone. With that clearly not being possible, I would like to begin by sending my sincere thanks to our reviewers, Asier and Istvan, who patiently provided us with feedback, comments, and creative criticism. Thanks, guys, for taking on this on top of your already hectic and busy day jobs to support me. Thanks also to our publisher, Pearson, and its amazing team who put up with our challenging schedules, updates, changes, and erroneous grammar. I would like to share a very special thank you to Meg Rainbow for her contribution on Chapter 1, Lee Sudduth for his contribution on Chapter 14, and Marcin Hamroz and Jaroslaw Gawron for their contributions on Chapter 22; these contributions added real-world perspectives from other vantage points that certainly enriched the overall manuscript. THANK YOU! Additionally, I would like to thank Kevin Regan, Alex Burger, Gino Corleto, Keith Baldwin, Mahesh Nagireddy, Jerome Dolphin, Sandeep Joseph, Einar Nilsen-Nygaard, Jonothan Eaves, Jeremy Cohoe, Kevin van Hengel, and Darrin Miller for their help and guidance along the way. To my leadership team (past and present), Adele Trombetta, Michael Kaemper, Markus Gierlich, and Haim Pinto, thank you for being supportive of my taking on this endeavor. And, last but certainly not least, I would like to thank my loving wife and children for their all-enduring support and motivation.

Dhrumil:

I would like to thank my leaders—Larry Hohmann, Mike Shomaker, and Jason Penn—for supporting me through this journey.

My peers, colleagues, and esteemed architects who have always shown their technical aptitude and willingness to help.

Lastly, to Amit Singh, without whom Chapter 12 of this book would not have been complete!

Ariel:

I would like to thank the great team at Pearson for their support in reviewing my content and supporting me along the way. I would also like to share a special thank you to my family and fiancé, Lilian, for their patience and support through the process of creating this book.

Vinay:

I would like to express my heartfelt gratitude to my management at Cisco for their unwavering support and encouragement, constantly motivating me to go beyond my defined role and explore new possibilities. Their guidance has been instrumental in shaping my professional growth.

I would also like to extend my appreciation to my colleagues, mentors, and the broader Cisco community for their collaboration, insights, and shared passion for innovation. Their support and camaraderie have made this journey truly enriching.

This book is a reflection of the collective inspiration and knowledge I have gained from working alongside such incredible individuals. Thank you for being a part of my journey.

Contents at a Glance

	Introduction	xxxix
Chapter 1	Zero Trust Demystified	1
Chapter 2	Secure Automation and Orchestration Overview	29
Chapter 3	Zero Trust Network Deployment	53
Chapter 4	Security and Segmentation	75
Chapter 5	DHCP and Dynamic Addressing Concepts	103
Chapter 6	Automating the Campus	127
Chapter 7	Plug-and-Play and Zero-Touch Provisioning	149
Chapter 8	Routing and Traffic Engineering	185
Chapter 9	Authentication and Authorization	219
Chapter 10	Quantum Security	253
Chapter 11	Network Convergence and Considerations	279
Chapter 12	Software-Defined Network Deployment Best Practices	315
Chapter 13	Wired and Wireless Assurance	337
Chapter 14	Large-Scale Software-Defined Network Deployment	361
Chapter 15	Cloud-Native Security Foundation	381
Chapter 16	Cloud-Native Application Security	437
Chapter 17	Data Center Segmentation On-Prem to the Cloud	487
Chapter 18	Using Common Policy to Enforce Security	535
Chapter 19	Workload Mobility: On-Prem to Cloud	571
Chapter 20	Resilience and Survivability	667
Chapter 21	Zero Trust in Industrial Manufacturing Vertical	691
Chapter 22	Third-Party SDN Integrations	717
Chapter 23	Infrastructure as Code (IaC)	745
	Index	777

Reader Services

Register your copy at www.ciscopress.com/title/ISBN for convenient access to downloads, updates, and corrections as they become available. To start the registration process, go to www.ciscopress.com/register and log in or create an account.* Enter the product ISBN 9780138204600 and click Submit. When the process is complete, you will find any available bonus content under Registered Products.

*Be sure to check the box that you would like to hear from us to receive exclusive discounts on future editions of this product.

Contents

	Introduction	xxxix
Chapter 1	Zero Trust Demystified	1
	Definition of Zero Trust	1
	How It All Began	2
	Why We Need Zero Trust	3
	Core Principles of Zero Trust	5
	Explicit Verification	5
	Least-Privilege Access	6
	Assume Breach	7
	Major Zero Trust Industry Standards	11
	People, Processes, and Technology	15
	People	16
	Processes	17
	Technology	18
	On-Premises vs. Cloud	19
	Explicit Verification	20
	Least-Privilege Access	21
	Segmentation	21
	Continuous Monitoring and Threat Detection	22
	Encryption and Data Protection	22
	Automated Response and Orchestration	22
	Endpoint Security	23
	Incident Response and Recovery	23
	Policy Enforcement	23
	Hybrid Environment Recommendations	23
	Security Certifications	24
	Summary	26
	References	27
Chapter 2	Secure Automation and Orchestration Overview	29
	Introduction to Automation and Orchestration	29
	Evolution of Network Automation	32
	Network Maturity for Automation	34
	Building Blocks of Secure Automation	35
	Secure Automation with API Security	35

	Dynamic Application Security Testing (DAST)	37
	Integrating XDR, SIEM, and SOAR	38
	Common Automation Practices and Tools	40
	Orchestration Using Ansible	44
	Orchestration Using Terraform	46
	CI/CD/CT	46
	AI and Machine Learning with Automation	47
	Machine Learning (ML)	48
	Neural Networks	49
	Generative AI and LLMs	50
	Data Lakes	50
	Summary	52
Chapter 3	Zero Trust Network Deployment	53
	Elements of Zero Trust Strategy Definitions	54
	Establishing Trust	55
	<i>User Trust Definition</i>	55
	<i>Device Trust Definition</i>	56
	<i>Trust Score Calculation</i>	58
	Defining Application Access Policies	59
	Enforcing Policies	60
	<i>Contextual Data</i>	61
	<i>Connection Metadata</i>	61
	<i>Logging Suspicious Actions</i>	61
	<i>Trust Tolerance</i>	62
	Tools and Technologies	63
	Central Inventory	63
	Identity and Access Management	63
	Network Segmentation	64
	Device Posture with Endpoint Security	65
	Virtual Private Network (VPN)	66
	Identifying Business Workflows	66
	Applying Zero Trust Using SSE	67
	Client-Based ZTNA Deployment for Managed Corporate Devices	68
	Clientless ZTNA Deployment for Unmanaged Devices	69
	VPN-Based ZTNA Using SSE	70
	SSE Integration for IoT Devices Using SD-WAN	70

	ZTNA Deployment Scenarios	71
	Greenfield ZTNA Deployment	71
	Brownfield ZTNA Deployment	73
	Summary	74
Chapter 4	Security and Segmentation	75
	Overview	75
	Segmentation Options	76
	Governance Considerations	77
	Macrosegmentation	77
	Routing Paths	80
	Stateful Inspection	81
	Audit Trail	81
	Failure Domain	82
	Microsegmentation	82
	Best Practices for Macro- and Microsegmentation	85
	Verification of Security Group Tags on IOS XE Platforms	88
	Methods of TrustSec Transport	91
	CTS Inline Tagging	92
	VXLAN Encapsulation	94
	GRE Encapsulation	94
	IPsec Encapsulation	95
	Control Plane TrustSec Transport	96
	SXP	96
	LISP	96
	Static	97
	SGT Priority Order	98
	Secure Service Insertion	98
	LAN-to-Cloud Microsegmentation	100
	Summary	101
Chapter 5	DHCP and Dynamic Addressing Concepts	103
	Introduction to Dynamic Addressing	103
	Zero Trust Approach to Dynamic Addressing	109
	Rogue DHCP Servers	109
	DHCP Starvation	112
	DHCP Man in the Middle	112
	DHCP Options	113

DHCP Authentication	114
IPv6 Address Assignment	115
<i>Well-Known Multicast</i>	117
<i>Transient Multicast IPv6 Addresses</i>	117
<i>Neighbor Discovery in IPv6</i>	117
<i>Solicited-Node Multicast Addresses</i>	118
<i>Anycast Addresses</i>	119
<i>Address Assignment in IPv6</i>	119
<i>DHCPv6 Options</i>	122
IPv6 First Hop Security	123
Rogue RA	123
DHCPv6 Guard	123
IPv6 Destination Guard	124
Source Guard and Prefix Guard	125
RA Throttle	125
ND Suppress Multicast	126
Summary	126
Chapter 6 Automating the Campus	127
Overview	127
Planning	128
IP Addressing	129
<i>Underlay Infrastructure IP Addresses</i>	130
<i>Management IP Addresses</i>	131
<i>Overlay User IP Addresses</i>	131
Maintaining IP Addressing Continuity	132
Site Hierarchy	135
Execution	135
LAN Automation	136
<i>LAN Automation Process and Workflow</i>	137
<i>API-Based LAN Automation Provisioning</i>	143
Partial Automated Deployment	144
Summary	147
References	147

Chapter 7	Plug-and-Play and Zero-Touch Provisioning	149
	Overview	149
	Plug-and-Play Provisioning	150
	Cisco Catalyst Center Call Flow	151
	Certificates	151
	Time Management	155
	Using IPv4 DHCP to Perform Plug-and-Play	156
	<i>DHCP Server Scope</i>	157
	<i>DNS Server: DHCP Option 6</i>	157
	<i>Domain Name: DHCP Option 15</i>	157
	<i>Vendor Class Identifier: DHCP Option 60</i>	157
	<i>DHCP Option 43</i>	159
	Using DNS to Perform Plug-and-Play	160
	Plug-and-Play Connect	161
	Startup VLAN	162
	LACP Usage with PnP	162
	Authorization of PnP Devices	163
	Meraki Onboarding Flow	164
	Zero-Touch Provisioning	165
	Foundation Configurations	165
	Software and Hardware Deployment Selection in Catalyst Center	166
	Claiming Devices in Catalyst Center	167
	Claiming Devices in the Meraki Dashboard	168
	Template Usage in Catalyst Center	169
	Standard Templates	169
	Composite Templates	170
	Bouncing Interfaces	171
	Programmability-Based Deployment	172
	Using Direct API Calls to Claim Devices	172
	Claiming Devices Using Ansible	177
	Customer Use Cases	177
	Large Banking Customer: Pan-Africa Deployment	177
	Global Deployment: Large-Scale Enterprise Deployment	180
	Summary	183

Chapter 8 Routing and Traffic Engineering 185

Overview	185
Routing	187
Underlay Routing Protocols	188
<i>Enhanced Interior Gateway Routing Protocol (EIGRP)</i>	190
OSPF	192
IS-IS	196
Overlay Routing Protocols	198
MPLS-VPN	200
BGP-EVPN	201
SD-Access	204
ACI	207
SD-WAN	209
Traffic Engineering	212
Network Design	213
Routing Protocols	214
Traffic Flow Analysis	214
Traffic Management	214
Load Balancing and Sharing	214
Quality of Service	215
Bandwidth Planning, Congestion, and Oversubscription	216
Network Monitoring and Optimization	216
Policy and Security	217
Global Internet	217
Geo-routing	218
Summary	218
References	218

Chapter 9 Authentication and Authorization 219

Overview	219
A Broader View of Identity	220
Authentication and Authentication Methods	223
Local Authentication	224
Centralized Server-Based Authentication	225
Service Accounts	228
When Using Service Accounts Goes Wrong	229
x.509 Certificate-Based Authentication	231
REST-API Authentication Methods	233

Multifactor Authentication (MFA)	234
Network Access Control	235
MAC Authentication Bypass	236
802.1x (Network Authentication)	240
Authorization	243
Dynamic Change of Authorization (CoA)	245
Identifying and Mitigating Risks of Unmaintained Virtual Machines in Network Access Control Deployments	246
Monitoring Authorization Health	248
Customer Use Cases	249
Using Dynamic Policy to Improve Real-World Challenges	250
Expediting System and Workstation Patches	250
Summary	252
Chapter 10 Quantum Security	253
What Is Quantum Computing?	253
The Need for New Computing Technologies	254
How Quantum Computing Is Different	255
Quantum Superposition	256
Qubit Modalities	257
Quantum Entanglement	258
<i>Gate Operations</i>	259
<i>H Gate</i>	261
<i>CNOT Gate</i>	262
<i>Current State</i>	264
Quantum Computing and Emerging Security Threats	265
Shor's Quantum Algorithm	268
Grover's Algorithm	269
Why Worry Now?	269
Approaches to Safeguard Against Quantum Adversaries	270
Symmetric Keys	270
Quantum Key Distribution	271
Practical Solution Approach	272
Quantum-Safe IPsec	273
Quantum-Safe MACsec	276
Dynamic Keys	276
Summary	278

Chapter 11 Network Convergence and Considerations 279

- What Is Convergence? 279
- Convergence in Layer 3 Routed Architectures 281
 - Protocol Convergence Timers 284
 - Server-Side Verification 288
- Convergence in Data Center Networks 290
- Convergence in Software-Defined Architectures 296
- Methodologies of Convergence Testing 300
 - Simulating Routed Convergence Scenarios 303
 - What Is Stateful Mode?* 304
 - What Is Stateless Mode?* 305
- Monitoring Security Convergence 308
- Summary 314

Chapter 12 Software-Defined Network Deployment Best Practices 315

- Introduction 315
- Network Deployment Lifecycle 317
- Stage 1: Planning and Design 318
 - Defining Network Requirements and Use Cases 318
 - Selecting the Right SDN Architecture 319
 - On-Premises Controller* 319
 - Software-as-a-Service (SaaS) Controllers* 320
 - Reviewing the Key Characteristics of SDN Controllers 321
 - Designing a Robust Network Topology 322
 - Addressing Security Considerations in SDN 323
 - Planning for Multidomain and Cloud Integration 324
- Stage 2: Deployment and Migration 324
 - Preparing the Network Infrastructure 325
 - Deploying the SDN Controller 326
 - Installing and Configuring the SDN Controller* 326
 - Setting Up High Availability and Disaster Recovery* 327
 - Prioritizing Network Services for SDN Migration* 328
 - Testing and Validating Migration Success* 329
- Stage 3: Operations and Management 330
 - Integrating SDN Automation with Existing IT Operations Management Systems 330

	<i>Monitoring and Troubleshooting the SDN Environment</i>	331
	<i>Maintaining Security and Compliance</i>	334
	Summary	335
	References	336
Chapter 13	Wired and Wireless Assurance	337
	What Is the Best Practice for Your Enterprise Architecture?	337
	Wired Network Best Practice Design Concepts	338
	Tiered Network Design	340
	Stacking Constructs	342
	Layer 3 Architectures	343
	Optimizing Wireless Networks	344
	Central Tunneling of Traffic (Over the Top)	350
	Local Breakout	350
	Anchoring Concepts (Catalyst/Meraki)	351
	Monitoring TrustSec and Security Enforcement	354
	Case Study: Financial Sector Customer	358
	Summary	360
Chapter 14	Large-Scale Software-Defined Network Deployment	361
	Introduction	361
	Network Design	362
	Physical Hardware: Bill of Materials	363
	Layer 2: Local Area Network	364
	Layer 3: Local Area Network	365
	Secure Connect: Wide Area Network	365
	Ordering and Delivery	366
	Security	367
	MX Security Features	367
	Security in Action	367
	ISE Integration	368
	Dynamic VLAN Assignment Case Study	368
	Adaptive Policy Microsegmentation	369
	Security Configuration	369
	Automation	369
	Meraki as Code	370
	Using Git for Configuration Management	371
	CI/CD Pipeline	373
	Fast Burger Automation	376

Implementation: Kyle and Jason Go to Fast Burger 377

Summary 379

Chapter 15 Cloud-Native Security Foundation 381

Introduction to Cloud-Native Security: A Zero Trust Perspective 381

From Cloud Infrastructure to Cloud Native: An Introduction to Cloud-Native Architectures 384

Characteristics of Cloud-Native Architectures: What Makes It Different? 384

Foundations of Cloud-Native Architectures 386

Containerization: The Building Blocks 386

Microservices Architecture: The Core of Modularity 387

Dynamic Orchestration and Management 387

Integrating DevOps and DevSecOps 388

Immutable Infrastructure and Scalability 389

Agility and Scalability: The Heartbeat of Cloud Native 389

Resiliency: Designing for Failure 389

API-Based Communication: Facilitating Service Interoperability 390

Observability: Insight into Cloud-Native Systems 390

Security: A Foundational Pillar 390

Building a Comprehensive Cloud-Native Security Strategy 390

Core Principles of Cloud-Native Architectures and Security 391

Why Microservices and Immutability Enhance Security 392

Service-to-Service Communication vs. Traditional Centralized Firewall 392

Cloud Infrastructure Security: Pillars and Practices in the Modern Cloud 393

The Shared Responsibility Model: A Foundation of Cloud Security 393

Architectural Foundations of Cloud Security in Hyperscaler Platforms 394

Unified Security Models and Identity and Access Management (IAM) 395

Data Encryption and Protection 395

Network and Infrastructure Security 396

Implementing Automated Compliance and Governance 398

Security Monitoring: Threat Detection and Response 398

Key Management in Cloud Environments 400

Understanding Key Management Systems (KMS) in the Cloud 400

Benefits of Using KMS 401

<i>Best Practices for Cloud-Based Key Management</i>	402
HashiCorp Vault: A Cloud-Native Key Management Solution	402
Components and Architecture of HashiCorp Vault	403
Network Security Evolution and Segmentation	404
Infrastructure as Code (IaC) and Security Automation	405
Advanced Load Balancing and Application Layer Security	406
<i>Application Load Balancers (ALBs): Features and Use Cases</i>	407
<i>The Future Landscape: Why OSS ALBs and Ingress Controllers Are Gaining Traction</i>	409
The Cloud-Native Security Stack: From Infrastructure to Application	411
Navigating Multicloud and Hybrid Cloud Security	413
Advanced Security Measures and Third-Party Services	414
The Need for Cloud Security Posture Management (CSPM)	415
<i>Features of CSPM</i>	415
<i>Benefits of CSPM</i>	415
<i>The Future of CSPM</i>	415
Integrating Cisco Solutions: Enhancing Multicloud and Hybrid Cloud Security with Attack Surface Management and JupiterOne	416
Cloud Workload Protection Platforms (CWPPs) and Cisco Secure Workload	417
Relationship to Zero Trust	419
Going Up the Stack from Infrastructure to Application	419
Monitoring and Logging Requirements for Compliance	421
Ensuring Visibility and Transparency Across the Cloud-Native Stack	422
Leveraging OpenTelemetry for Security	422
Splunk for Enhanced Security Posture	423
Continuous Monitoring and Automation Regulatory Requirements and Compliance Standards	424
Emerging Trends and Technologies in Cloud-Native Security	425
The Role of AI and ML in Enhancing Security Postures	425
<i>Enhancing Security with AI/ML: A Practical Cisco Scenario</i>	426
<i>Anticipating Future Threats and Preparing Defenses</i>	426
Embracing Continuous Threat Exposure Management (CTEM) for Enhanced Cybersecurity	427
<i>More Than Vulnerability Management Evolution: Why CTEM Is Gaining Traction</i>	428
<i>Getting Started with CTEM</i>	429

The Evolving Landscape of Cloud-Native Security Standards and Framework	430
Incorporating Matured Zero Trust Frameworks into Cloud-Native Security	431
<i>DISA Zero Trust Framework</i>	431
<i>CISA Zero Trust Maturity Model V2.0</i>	432
<i>NIST Special Publication 800-207—Zero Trust Architecture</i>	432
<i>Cisco Zero Trust Framework</i>	432
The Role of Zero Trust Frameworks in Evolving Cloud-Native Security	433
The Cisco SAFE Security Reference Model	434
Summary	435
References	436

Chapter 16 Cloud-Native Application Security 437

Introduction to Cloud-Native Application Security	437
Definition and Scope	438
Key Challenges	438
Evolution from Traditional Security	439
Understanding OWASP and Cloud-Native Security Risks	439
<i>OWASP Top 10: Web vs. Cloud-Native</i>	440
<i>Expanding OWASP Principles for Cloud-Native Architectures</i>	444
<i>Should We Consider a Cloud-Native-Specific Model?</i>	445
CNCF Projects for Cloud-Native Security	445
1. <i>Provisioning</i>	446
2. <i>Runtime</i>	446
3. <i>Orchestration and Management</i>	447
4. <i>App Definition and Development</i>	447
5. <i>Observability and Analysis</i>	448
6. <i>Platforms</i>	448
<i>Top 20 CNCF Projects Focused on Security</i>	450
<i>Cloud</i>	453
<i>Clusters</i>	454
<i>Containers</i>	456
<i>Code</i>	457
Role of Cloud-Native Application Protection Platform (CNAPP)	458
API Security	459

Vulnerability Management	459
Runtime Protection	459
Policy Enforcement	459
Compliance Management	460
Building Secure Applications with Cloud-Native Security	460
Security in Application Design and Development	460
<i>Principle of Least Privilege</i>	460
<i>Immutable Infrastructure</i>	461
<i>Secrets Management</i>	461
<i>Secure Coding Practices</i>	462
Shift-Left Security and DevSecOps Integration	463
<i>DevSecOps: Embedding Security into DevOps</i>	463
<i>Top Cloud Security Risks in DevOps</i>	463
<i>Strategies for Enhancing Collaboration</i>	463
Managing Dynamic Cloud Configurations	464
<i>Configuration Management Tools</i>	464
<i>Continuous Monitoring and Drift Detection</i>	464
<i>Dynamic Secrets Management</i>	464
<i>Automated Remediation Tools</i>	464
<i>Infrastructure as Code (IaC) and Security</i>	465
<i>Securing the Software Supply Chain</i>	465
API Security	467
<i>Secure APIs Using Kubernetes Security</i>	468
Unique Security Considerations for Serverless Architectures	470
Serverless Shared Responsibility Model	471
Key Serverless Security Challenges	471
<i>Function-Level Permissions</i>	471
<i>Dependency Management</i>	472
API Gateway Security	472
<i>Visibility and Monitoring</i>	472
<i>Proactive Threat Detection and Response</i>	472
<i>Data Leakage</i>	472
The Path Forward for Serverless Security Best Practices	472
<i>Enforce Fine-Grained Permissions</i>	473
<i>Manage Dependencies Proactively</i>	473
<i>Harden API Gateways</i>	473

<i>Enhance Visibility and Real-Time Monitoring</i>	473
<i>Protect Sensitive Data and Logs</i>	474
<i>Automate Security Validation and Compliance</i>	474
<i>Proactive Threat Detection and Response</i>	474
<i>Collaborative Security with DevSecOps</i>	474
<i>Adopt a Security-First Mindset with Automation</i>	475
Critical Attack Vectors in Serverless Applications	475
<i>Function Input Manipulation</i>	475
<i>Access Control and Authentication</i>	476
<i>Resource and Configuration Attacks</i>	476
<i>Event Injection</i>	476
<i>Cold Start Abuse</i>	476
<i>Other Common Security Risks Witnessed</i>	476
Detailed Security Flow Through Components	477
1. <i>User Request Initiation</i>	478
2. <i>API Gateway Processing</i>	479
3. <i>Function Trigger Evaluation</i>	479
4. <i>Permission Check Verification</i>	479
5. <i>Function Execution Management</i>	480
6. <i>Monitoring Layer</i>	480
7. <i>Data Services Integration</i>	481
8. <i>Security Scanning</i>	481
<i>Database Access Example</i>	482
Emerging Trends and Future Outlook in Cloud-Native Security	482
Reimagining Cloud Security and Zero Trust with CNAPP Solutions	483
Toward Proactive Cloud Security with CNAPP Solutions	483
Enhancing API Security with LLMs and CNAPP Solutions	484
Summary	485
References	486

Chapter 17 Data Center Segmentation On-Prem to the Cloud 487

Introduction to Data Center Segmentation in Hybrid and Multicloud Environments	487
The Limitations of Traditional Segmentation	488
A New Approach to Segmentation	488
Zero Trust and Microsegmentation Principles for Segmentation	489
What Is Zero Trust?	489
Key Benefits of Zero Trust	490

The Synergy of Zero Trust and Microsegmentation	490
Segmentation Challenges in Hybrid and Multicloud Environments	491
Inconsistent Policy Frameworks	492
Visibility and Compliance	492
Dynamic Workloads	492
Ways to Address These Challenges	493
Ways to Implement End-to-End Segmentation Policies with Zero Trust	493
Unified Policy Definition Across Domains	494
Workload-Specific Segmentation	494
Methods to Prevent Policy Drift	494
Core Features of Cisco Cloud Network Controller for Unified Segmentation	495
The Role of Zero Trust in Unified Policies	495
Ways to Migrate Segmentation Policies: From On-Premises to Cloud	496
Adapting Segmentation to Cloud-Native Architectures for Zero Trust Integration	496
Navigating the Transition from Cisco Cloud Controller	497
Overview of Policy Models: Cisco ACI and Cloud-Native Constructs	498
<i>Cisco ACI Policy Model</i>	498
<i>Cloud Policy Models (AWS, Azure, GCP)</i>	499
<i>Comparison and Mapping of Policy Constructs</i>	500
<i>Comparison to Cloud Segmentation Models</i>	501
Contracts as Security Policy Objects in Cisco ACI	501
<i>Provider-Consumer Model in Cisco ACI</i>	502
<i>How Cisco ACI Contracts Simplify Network Management</i>	503
Effective Segmentation Migration Methods	505
<i>Phased Migration with Policy Layers</i>	505
<i>Parallel Deployment for Hybrid Continuity</i>	505
<i>Automated Policies for Rapid “Big Bang” Migrations</i>	506
Consistency Across Hybrid and Multicloud Environments	506
<i>How to Leverage Tenants for Multisite Management</i>	507
<i>Consistent Policy Enforcement and Centralized Management</i>	507
<i>Dynamic Policy Enforcement and Adaptation</i>	507
<i>Operational Efficiency and Automation</i>	508
<i>Multisite Application of Tenants</i>	508
<i>How to Use a Multicloud Policy Orchestrator Across Clouds</i>	508

Open-Source Cross-Domain Orchestration and Automation for Segmentation Policies	510
<i>Integration of Open-Source Tools for Cross-Cloud Policy Consistency</i>	510
<i>AI/ML for Adaptive Policy Enforcement in Hybrid Cloud Environments</i>	511
<i>Visibility, Monitoring, and Predictive Analytics in Hybrid or Multicloud Environments</i>	513
Web3 and Immutable Trust in Hybrid Cloud Segmentation	514
What Is Web3?	514
Web3 Technologies and Zero Trust: A Symbiotic Relationship	514
<i>How Decentralized Identity (DID) Works</i>	515
<i>Hybrid Cloud Challenges and Web3's Role in Policy Enforcement</i>	516
<i>Decentralized Identity: A Cornerstone of Web3 Security</i>	516
<i>Ways to Enhance Security with Blockchain's Immutability</i>	517
<i>Security in Web3 and Smart Contracts</i>	517
<i>Blockchain for Immutable Access Logs</i>	517
<i>Web3's Value Beyond Blockchain: A New Security Paradigm</i>	518
DID in Cloud-Native and Hybrid Environments	518
<i>Real-World Applications of DID and Zero Trust</i>	519
<i>Workflow: Kubernetes Access Using DID and Smart Contracts</i>	519
<i>Coding Mistakes and Exploits</i>	520
<i>Latency Concerns</i>	521
<i>Smart Contract Revocation Challenges</i>	522
<i>Kubernetes Webhook Security Risks</i>	522
<i>Blockchain Cost and Scalability</i>	522
Integrating Kubernetes with Blockchain for Smart Contract–Based Access Control	523
<i>Decentralized Identity Authentication in Kubernetes with Keycloak SPI</i>	523
<i>Authentication Flow Implementation Details for DID and Smart Contracts</i>	523
<i>User Initiates Authentication (Signing Challenge with Web3 Wallet)</i>	523
<i>Keycloak SPI Validates Signature and DID</i>	524
<i>The OIDC Token Is Issued</i>	524
<i>Kubernetes Validates the OIDC Token</i>	525
<i>Developer Uses Token to Access Kubernetes</i>	525

	<i>Kubernetes Enforces Access Using Smart Contracts Authorization</i>	526
	<i>External Admission Controller (Validating Webhook)</i>	527
	<i>Kubernetes Enforces DID-Based Segmentation</i>	529
	<i>Smart Contracts for Policy Enforcement</i>	530
	<i>Solidity Contract Example</i>	530
	Audit Logging: Immutable Developer Access Logs	532
	Benefits of This Approach	533
	Summary	534
	References	534
Chapter 18	Using Common Policy to Enforce Security	535
	Introduction to Security Policies	535
	What Is a Cloud Security Policy?	536
	Principles of Effective Policy Management	536
	Designing Common Security Policies	536
	Unifying Security Policy Frameworks	537
	Balancing Granularity and Manageability	537
	Standardizing Policies Across Diverse Environments	537
	Creating Consistency Across Environments	537
	Adapting to Changing Access Patterns with Common Policy	538
	Policy Enforcement Mechanisms	539
	Firewalls and Intrusion Detection/Intrusion Prevention Systems (IDS/IPS)	539
	Cloud Access Security Brokers (CASBs)	539
	The Role of CASB in Cloud Security	539
	Security Orchestration, Automation, and Response (SOAR)	540
	CASB and SOAR: A Harmonious Approach	541
	Identity and Access Management (IAM) Policies	541
	Crafting Consistent IAM Policies	541
	Role of Identity Federation and Single Sign-On (SSO)	541
	Managing Privileged Access	542
	Implementing Multicloud IAM	542
	Data Protection and Privacy Policies	543
	Aligning Data Governance with Security Policies	543
	Ensuring Compliance with Regulations	543
	Network Security Policies	543
	Segmentation Policies for Network Security	544
	Secure Network Configuration and Management Policies	544

From SDLC to SDL to SSDLC: A Journey Toward Secure Software Development	544
Software Development Life Cycle (SDLC): The Foundation of Application Development	544
<i>Origins of SDLC</i>	544
<i>Phases of SDLC</i>	545
Security Threat and Vulnerability Assessment and Measurement in Secure Software Development	545
<i>Common Vulnerability Causes</i>	545
<i>Taxonomy in Action Example</i>	546
Transitioning from SDLC to SDL: Embedding Security	547
<i>SDL Enhancements to SDLC Phases</i>	548
<i>Why the Enhancements of SDL Are Beneficial</i>	549
Secure Software Development Lifecycle (SSDLC): Evolving with Agile and DevOps	550
Key Enhancements in SSDLC	550
Phases in SSDLC	552
Benefits of SSDLC	552
Understanding the Evolution of Software Development Security Frameworks	553
<i>Key Transition Milestones</i>	553
<i>Framework Comparison</i>	555
OWASP SAMM: A Framework for Security Maturity	557
Understanding SAMM's Structure	557
<i>SAMM Framework Overview</i>	558
<i>Governance: A Business Function in Focus</i>	559
<i>How SAMM Relates to SDLC, SDL, and SSDLC</i>	561
<i>Benefits and Implementation</i>	562
Monitoring, Logging, and Auditing Policies	563
Unified Logging and Continuous Monitoring	563
Automated Auditing and Compliance Reporting	563
Incident Response and Remediation Policies	564
Incident Response Frameworks	564
Automated Policy-Based Remediation	564
Policy Compliance and Verification	564
Continuous Compliance Monitoring	564
Integrating Policy Checks into CI/CD Pipelines	564
Challenges in Policy Enforcement Across Hybrid Environments	565

Inconsistent Enforcement Capabilities	565
The Need to Overcome Resistance to Unified Frameworks	565
Future Directions in Policy-Based Security	565
Predictive and Adaptive Policies	565
The Role of AI in Dynamic Policy Management	565
Security Suites Delivered by the Cisco Security Cloud	566
<i>Cisco User Protection Suite</i>	566
<i>Cisco Cloud Protection Suite</i>	567
<i>Cisco Breach Protection Suite</i>	568
Summary	568
References	569

Chapter 19 Workload Mobility: On-Prem to Cloud 571

Definition and Scope of Workload Mobility	571
Is Your Cloud Ready for Your Workloads? Understanding the Benefits and Challenges	572
Real-World Application: ABC Corp's Cloud Migration Journey	574
Motivations for Cloud Migration	574
<i>Cost Efficiency and Budget Management</i>	575
<i>Innovation-Friendly and Fast Development Environment</i>	575
<i>Access to Advanced Computing Capabilities</i>	575
<i>Enhanced Security and Compliance</i>	575
<i>Focus on Core Business Activities</i>	576
<i>Global Reach and Market Expansion</i>	576
Hybrid IT Infrastructure and Workload Placement	576
<i>On-Premises Infrastructure</i>	576
<i>Multicloud Strategy</i>	576
<i>Co-Location and Its Role in Modern IT Strategies</i>	577
<i>The Deployment of Cloud Smart on Hybrid IT Environments</i>	578
<i>Cloud Smart Environment for Workload Placement</i>	578
Choosing a Cloud Model with Zero Trust as the Goal	579
Infrastructure as a Service (IaaS)	579
Platform as a Service (PaaS)	580
Software as a Service (SaaS)	580
Containers as a Service (CaaS)	580
Function as a Service (FaaS)	581
Analysis of TCO and ROI for Workload Migration	581

Migration Context	581
Moving Applications and Data	582
Deployment Model	582
People, Processes, and Technology	582
Building Out a Secure Migration Plan	583
Migration Strategies: From the Five to Seven R's	583
Security Considerations for Data Transfer and Transition Phases	586
Integrating AWS's Well-Architected Framework: Case Study of ABC Corp	587
Building a Strong Identity Foundation	587
Maintaining Traceability	587
Applying Security at All Layers	587
Making Security Best Practices into Habits	587
Ensuring Data Is Protected in Transit and at Rest	588
Keeping People Away from Data	588
Preparing for Security Incidents	588
Performing Risk Assessment and Mitigation	588
Embracing the AWS Shared Responsibility Model	588
Connecting with the Well-Architected Frameworks	588
Workload Migration Frameworks and Tools	589
Leveraging Migration Services and Tools	590
Understanding the Role of Automation and Integration	590
Ensuring Data Security During Migration	591
Optimizing Cloud Migration Outcomes	591
<i>The "What" to Migrate</i>	591
<i>Comprehensive Tooling for Migration</i>	592
Data Security During Workload Migration	593
Ensuring Data Integrity and Confidentiality	593
What Is Secure Data Transfer?	594
<i>Types of Cloud Data Transfer</i>	595
<i>Online Data Migration</i>	595
<i>Network Data Transfers</i>	595
<i>Direct Connect Services</i>	596
<i>Hybrid Data Migration</i>	597
<i>Offline Data Migration</i>	597
<i>Considerations for Each Approach</i>	597
Secure Data Transfer Best Practices	597

Data Transfer vs. Cloud Migration: An Overview	598
Data Transfer Considerations as Part of a Broader Cloud Migration Strategy	599
<i>The Impact of Bandwidth Limitations</i>	599
<i>Tactics for Mitigating Bandwidth Limitations</i>	600
Data Security Concerns	600
Downtime Risks	601
Compatibility Issues	601
<i>Legacy Systems vs. Modern Cloud Infrastructure</i>	601
<i>Data Format and Structure</i>	601
<i>Use Cases in Cloud Migration</i>	602
<i>Tools and Their Application in Cloud Migration</i>	603
<i>Application Compatibility</i>	603
Cloud Migration Security	604
What Makes Migration to the Cloud Such a Necessity?	604
What Is Cloud Migration Security?	605
Risks of Cloud Migration	605
Preparing for Cloud Migration Security Concerns	606
<i>API Vulnerabilities</i>	606
<i>Security Blind Spots</i>	606
<i>Compliance Requirements</i>	606
<i>Data Loss</i>	607
Effective Safeguarding for Executing a Cloud Migration	607
<i>Understanding Your Data and Compliance Requirements</i>	607
<i>Safeguarding APIs and Access Controls</i>	607
<i>Encrypting Your Data During Transit</i>	608
<i>Limiting Data Access During Cloud Migration</i>	608
<i>Employing a Phased Migration Strategy and Risk Mitigation</i>	608
<i>Implementing Decommissioning and Sanitization Activities</i>	609
<i>Formulating a Security Plan</i>	609
<i>Maintaining Data Protection and Integrity</i>	609
<i>Confirming Security Measures</i>	610
Cloud Migration: Security Checklist	610
<i>Pre-Migration Security Preparations</i>	610
<i>Security During Migration</i>	612
<i>Post-Migration Security Maintenance</i>	613
Quality Engineering: The Heart of Cloud Migration	614

<i>The Landscape of Cloud Migration Challenges</i>	614
<i>Quality Engineering's Strategic Blueprint</i>	614
<i>Ensuring Data Security and Navigating Cloud Migration with Precision</i>	615
Network and Connectivity Considerations	616
The Maintenance of Network Security Postures During and After Migration	616
Continuous Monitoring and Assessment	616
Access Controls and Segmentation	618
<i>Network Segmentation</i>	618
<i>Access Controls</i>	619
Network Considerations for Cloud Migration	621
Measuring and Ensuring Network Performance	621
Consistently Measuring Network Performance	621
Measuring and Assessing Network Performance by Monitoring Key Metrics	623
Network Performance Measurement Tools	624
Application Dependency Mapping (ADM) and Application Performance Management (APM): Navigating Secure Cloud Migration and Zero Trust Architecture	625
<i>ADM: Charting the Cloud Migration Terrain</i>	625
<i>APM: The Compass for Cloud Optimization</i>	626
<i>ADM and APM: Synergizing Security and Performance</i>	626
Leveraging Hyperscaler Observability Tools for Strategic Cloud Integration	626
Core Benefits of ADM and APM Integration in Dynamic Cloud Environments	627
Integrating Observability into Cloud Migration	628
<i>Key Outcomes</i>	629
<i>Metrics and Monitoring</i>	629
The Role of OpenTelemetry and CNCF in Cloud Migration	630
Integrating OpenTelemetry into Network Measurements	632
<i>Integration Approaches</i>	632
<i>A Real-World Example for Integrating OpenTelemetry into the Networking Domain</i>	634
Tying It All Together	634
End-to-End Visibility on Digital Experience Using Cisco ThousandEyes	635
<i>The Prologue: Setting the Stage for Migration</i>	635

<i>The Journey: Navigating the Migration</i>	636
<i>The Finale: Validating Success</i>	636
Managing IP Addressing and DNS Changes	637
Challenges in IP Planning	638
Best Practices for IP Planning	638
Designing Virtual Networks	639
Universal Best Practices Across Cloud Platforms	639
Best Practices for Virtual Networks Designs and Considerations During Cloud Migration	640
DNS	642
Ensuring High Availability and Disaster Recovery Readiness	643
Selecting a Cloud Provider and Service Model	644
Designing Your Cloud Architecture and Migration Strategy	644
Executing and Monitoring Migration and Disaster Recovery	645
Security Posture Adjustment Post-Migration	645
Adjusting Security Controls to the Cloud Environment	646
Monitoring and Responding to Security Events During the Stabilization Period	646
Updating Incident Response and Forensic Capabilities	647
Using Hyperscalers and Cisco-Provided Security Tools	647
Identity and Access Management in Hybrid Environments	649
IAM Basics	649
Significance of IAM in Hybrid Environments	650
Strategic Considerations for IAM in Hybrid Environments	650
IAM Role Mapping and Policy Enforcement	651
Federated Identity for Seamless Access Control	651
Privileged Access Management in a Hybrid Setting	651
Key IAM Migration Considerations	652
Best Practices and Considerations for IAM Implementation	653
How to Structure Resources and Permissions in the Cloud	653
<i>Permissions and IAM</i>	654
<i>Explanation and Context</i>	656
<i>External Access and Security Considerations</i>	657
<i>Guardrails and Automated Analysis</i>	661
<i>Automated Analysis Solutions</i>	663
Summary	664
References	665

Chapter 20 Resilience and Survivability 667

- Resilience Metrics 667
- Types of Resilience 671
 - Physical Resilience 671
 - Environmental Redundancy 672
 - Domain Resilience 672
 - Vendor Resilience 673
- Software Resilience 674
 - Software Versioning 674
 - In-House vs. Commercial Off-the-Shelf (COTS) Software 675
- Resilience in the Cloud 676
- Consequences of Authentication and Authorization Resilience 681
- Client and Server Agent Resilience 684
- Audit Trail Resilience 686
 - Audit Trail Reputability 687
 - Reliability of Auditable Data 688
- Proactive Resilience Validation 689
- Network Infrastructure Resilience Consideration 690
- Summary 690

Chapter 21 Zero Trust in Industrial Manufacturing Vertical 691

- Introduction to Industrial Networking 691
- Pillars of ZTNA for Industrial Plant Networks 696
 - Security Foundation with Firewalls 696
 - Visibility with the Network as a Sensor 698
 - Creating Granular Trust Zones Using Microsegmentation 702
 - Use Case 1* 704
 - Use Case 2* 704
 - Correlation and Automation to Contain Threats 705
- Secure Remote Access with ZTNA 706
- Extending ZTNA in a Noncarpeted Environment with Cisco SD-Access 710
 - Extended Node (EN) 711
 - Policy Extended Node (PEN) 712
- Summary 715

Chapter 22	Third-Party SDN Integrations	717
	Introduction to Third-Party SDN Integrations	717
	End-to-End Policy Strategy in a Multivendor Environment	718
	Benefits of End-to-End Segmentation	718
	Challenges in Multivendor Environments	719
	Scenario 1: Site Type A with Cisco SD-WAN Integration	721
	Scenario 2: Site Type B with SD-WAN Integration	721
	Scenario 3: Site Type C with SD-WAN Integration	722
	Scenario 4: Inter-Regional Communication	723
	Why VXLAN-EVPN?	723
	BGP EVPN Detailed Traffic Flow and Architecture	725
	Security Considerations in the Campus	727
	Firewall Connectivity in the Campus	728
	Third-Party Vendor Firewall Policy Integration	735
	Highly Resilient Firewall Integrations	740
	Summary	743
	References	743
Chapter 23	Infrastructure as Code (IaC)	745
	Introduction	745
	Evolution of Automation in Network Device Deployment and Management	746
	Working with Structured Data	758
	Revision Control	761
	Building a Data Model	764
	Network Controllers vs. Direct to Device	765
	Deploying an IaC Architecture	766
	Securing IaC Provisioning	769
	NETCONF Service-Level Restrictions	769
	Deploying a Resilient “as Code” Infrastructure	772
	“As Code” Today	773
	Transitioning to a Network “as Code”	774
	Pre-Validation in the Physical Replica or a Digital Twin	775
	Summary	776
	Index	777

Icons Used in This Book



Users



Laptop



Smartphone



Printer



PCs



Router



Switch



Cloud



Layer 3 Switch



Firewall

Identity Services
Engine (ISE)Cisco Nexus
7000Wireless LAN
Controller

Command Syntax Conventions

The conventions used to present command syntax in this book are the same conventions used in the IOS Command Reference. The Command Reference describes these conventions as follows:

- **Boldface** indicates commands and keywords that are entered literally as shown. In actual configuration examples and output (not general command syntax), boldface indicates commands that are manually input by the user (such as a **show** command).
- *Italic* indicates arguments for which you supply actual values.
- Vertical bars (|) separate alternative, mutually exclusive elements.
- Square brackets ([]) indicate an optional element.
- Braces ({ }) indicate a required choice.
- Braces within brackets ({{ }}) indicate a required choice within an optional element.

Introduction

The idea for this book originally came into existence during a time that the four authors were heavily involved in global deployments of large and complex networks focused on zero trust architectures. Pre-COVID, many of the authors were traveling up to 42 weeks a year, meeting with customers, supporting large deployments onsite, and hearing first-hand about the challenges, pitfalls, and gotchas they were experiencing from deploying advanced and new technology suites that could lead to a more robust and secure network architecture.

Over time, the team observed the technology mature, the escalations and teething issues seen in the early days of deployments of zero trust networks subside, and customers shift gears to realize their visions of a more automated, dynamic, and secure estate, focusing on the principles of providing only the requisite access for a given service. In addition to the maturity of the technology, the expansion of domains to adopt zero trust principles moved beyond the campus, to other domains and verticals, such as edge compute, the data center, cloud, and deployment in containerized form factors in AI agents. These adoptions further cemented the need to ensure a robust and secure estate, regardless of the location of the user, endpoint, workload, or autonomous/semi-autonomous software process.

The key focus areas of this book are not specific to a single technology; this title provides a focus on the domains of resilience, automation, security, and cloud architectures. While separate in their own right, when utilized together, they can result in a scalable, secure, and future-proof deployment.

Who Should Read This Book?

This book is aimed at security engineers, security analysts, site reliability engineers, network engineers, architects, and operations teams who have a key focus on optimizing their network deployments, increasing security, resilience, and automation within their organization. Given the experience of the authors, a heavy focus has been placed on real-life examples from their years of deployment experience globally. While it would have been easy to describe the solutions through rose-colored lenses, this book provides a candid overview of the good, the bad, and the ugly that you can experience on your journey toward building an automated, resilient, and secure enterprise estate.

How This Book Is Organized

Chapter 1, “Zero Trust Demystified”: This chapter provides an overview of the importance of zero trust in an organization’s network.

Chapter 2, “Secure Automation and Orchestration Overview”: This chapter provides an overview of the automation and orchestration strategies available during the writing of this book.

Chapter 3, “Zero Trust Network Deployment”: This chapter provides an overview of zero trust deployment on secure service edge and other architectures.

Chapter 4, “Security and Segmentation”: This chapter takes you on a deep dive into micro- and macrosegmentation concepts and explains where they can be best applied for a modern enterprise network.

Chapter 5, “DHCP and Dynamic Addressing Concepts”: This chapter provides an overview of dynamic addressing and security concepts related to its use.

Chapter 6, “Automating the Campus”: This chapter covers concepts of campus automation and how to achieve them securely.

Chapter 7, “Plug-and-Play and Zero-Touch Provisioning”: This chapter describes aspects of secure zero-touch and plug-and-play onboarding of network devices in enterprise domains, including resilience automation aspects.

Chapter 8, “Routing and Traffic Engineering”: This chapter covers concepts of routing and traffic engineering, showing how proper planning and foresight can lead to building the most resilient networks.

Chapter 9, “Authentication and Authorization”: This chapter details concepts associated with authentication and dynamic authorization in enterprise architectures.

Chapter 10, “Quantum Security”: This chapter explains the deployment of quantum-based security and explores the concepts associated with how quantum security is relevant to a network estate.

Chapter 11, “Network Convergence and Considerations”: This chapter explores concepts of convergence in software-defined architectures for enterprise, data center, and security domains in detail.

Chapter 12, “Software-Defined Network Deployment Best Practices”: This chapter covers the best practices of SDN networks and how to deploy them.

Chapter 13, “Wired and Wireless Assurance”: This chapter explores enterprise wired and wireless deployment constructs from a resilience perspective, including security options for guest network access in Cisco cloud-based Meraki and on-premises deployments.

Chapter 14, “Large-Scale Global Software-Defined Network Deployment Best Practices”: This chapter focuses on a global Meraki network deployment, whereby Infrastructure as Code concepts and approaches are used to simplify and scale the deployment.

Chapter 15, “Cloud-Native Security Foundation”: This chapter covers key security principles for cloud-native environments, focusing on zero trust, workload protection, IAM, encryption, and compliance. It also explores CSPM, infrastructure as code (IaC) security, and AI-driven threat detection.

Chapter 16, “Cloud-Native Application Security”: Focusing on securing applications in dynamic environments, this chapter covers DevSecOps, CI/CD security, API protection, and OWASP best practices. It also explores Web3-based identity (DID) and AI-driven security automation.

Chapter 17, “Data Center Segmentation On-Prem to the Cloud”: This chapter examines segmentation strategies for hybrid and multi-cloud environments, highlighting zero trust, microsegmentation, and policy enforcement. It also explores cloud-native segmentation models and blockchain-based security.

Chapter 18, “Using Common Policy to Enforce Security”: This chapter discusses unified security policies, IAM best practices, and automated enforcement using CASBs and SOAR. It also covers software security frameworks (SDLC to SSDLC) and OWASP SAMM for security maturity.

Chapter 19, “Workload Mobility: On-Prem to Cloud”: This chapter explores workload migration strategies, zero trust integration, data security, and compliance. It also covers post-migration optimization, including cost management and observability tools.

Chapter 20, “Resilience and Survivability”: The chapter focuses on key concepts around redundancy, survivability, and resilience for network architectures in the context of routed network deployments and security architectures.

Chapter 21, “Zero Trust in Industrial Manufacturing Vertical”: This chapter explores OT/IoT-type deployments in the domain of security and zero trust.

Chapter 22, “Third-Party SDN Integrations”: This chapter explores the integration of third-party devices and systems into a broader zero trust architecture and domain.

Chapter 23, “Infrastructure as Code”: This chapter describes key concepts associated with deploying automation for Infrastructure as Code in secure enterprise networks.

Credits

Front Cover - TechAnimationStock/Shutterstock

Figures 4.12, 4.14, 7.2, 9.13, 11.25 & 13.7 - Wireshark Foundation

Figure 11.21 - Keysight Technologies

Figure 15.2 - Center for Internet Security

Figure 15.6 - Gartner, Inc

Figures 16.1 & 23.3 - GitHub, Inc

Figure 19.1 - Flexera

Figure 19.02- Amazon Web Services, Inc

Figure 19.3 - ChaosSearch, Inc.

Zero Trust Network Deployment

In this chapter, you will learn about the following:

- Zero trust and functional pillars
- Elements of zero trust policy
- Tools and technologies for zero trust deployment
- Greenfield versus brownfield zero trust network deployment

In today's risky and uncertain times, organizations need to build a solid foundation for reliable and adaptable security operations. This could be achieved by building zero trust in their IT systems. The zero trust framework operates on the principle of “never trust, always verify.” Unlike traditional security approaches that rely on a strong perimeter (like firewalls) to protect a trusted internal network, zero trust assumes that threats can originate both inside and outside the organization. Therefore, every request, user, or device must be authenticated and authorized, regardless of its location within or outside the network. It is important to understand the functional pillars of zero trust before any organization starts deploying it. You will need a variety of tools and platforms to deploy zero trust in your organization. Your zero trust approach must be able to perform the following four functions:

- **Establish Trust:** Ensure the system can verify each user and device connecting to the network.
- **Enforce Trust-Based Access:** The system should be able to enforce policies and grant access to the users and devices based on the trust level. The policies should be applied based on the principle of least privilege. The access mechanism should protect all network resources, including applications and data, on-premises and in the cloud.

- **Verify Trust:** The system should be able to continuously verify the trust and dynamically adjust the access to the network. It should not be a time check and then access is allowed for a longer duration or to multiple applications.
- **Enforce Policies and Monitor:** The system should be able to react to changes in trust by analyzing and coordinating responses to potential incidents while gaining better visibility into suspicious activities related to trust levels. A continuous refinement of trust policies is required.

An organization needs to protect all its assets irrespective of their location. Some assets could be mobile-like users and devices, and others like applications, and stationary databases could either be on-premises or hosted in the cloud. The zero trust approach should consider various factors, such as user type, location, device type, data requested, and application accessed, to create robust dynamic policy-based access controls. Some of the common assets you want to protect in your organization include (but are not limited to)

- Users
- Devices
- Network
- Cloud
- Applications
- Data

A common question arises: Where to start, and how can you define the steps toward a zero trust framework adoption?

In the rest of this chapter, you will learn the iterative approach that could be adopted for securing trusted access for users and devices based on the four primary functional requirements of zero trust as covered in this section. We will first look at the key decision points to establish the overall zero trust strategy that includes establishing trust for users and devices, application access, and policy enforcement. Next, we will look at the common tools and technologies to help you deploy zero trust, and finally, we will look at the approach for zero trust adoption in greenfield and brownfield scenarios.

Elements of Zero Trust Strategy Definitions

In this section, we will look at the key considerations that you will need to formulate the zero trust deployment policy. These considerations include

- Establishing user trust
- Establishing device trust
- Defining application access policies
- Enforcing policies

Establishing Trust

A trust level represents the degree of confidence you have that a user or device is who or what they claim to be and that they are authorized to access specific resources. Unlike older models, where access is granted based on location (inside vs. outside the network), trust levels in zero trust are dynamic and can change based on several factors. Trust levels can be assigned to users and devices.

- **User Trust Level:** Based on the identity of the person accessing the network
- **Device Trust Level:** Based on the security status of the device accessing the network

The first functional requirement of zero trust is to establish trust for users and devices. This requirement could be divided into two categories: users and devices. Let's look at both of them in detail.

User Trust Definition

The purpose of defining user trust is to verify the user's trust and enforce the principle of least privilege. You need to start evaluating what mechanisms and processes are there in your organization to authenticate and authorize the users before they can access the resources. But wait, which users? There could be different types of users having different privilege access. To start, do the following:

1. Identify the user types and roles (employees, including contractors, temporary users, and guests).
2. Associate the risk with each user type, based on their function and type of data they need to access.
3. Identify the assets that need to be accessed by these user groups.
4. Identify the location of assets each group needs to access—on-premises, cloud, DMZ, and so on.

Following these steps, you will get a matrix that shows the user group and their need to access, which type of information, and where that information is stored. As a next step, you need to define the trust level for each user type based on the authentication and authorization system in place or planned. You can start evaluating as follows:

- How does an employee need to be authenticated and authorized?
- How does a partner need to be authenticated and authorized?
- Can you deploy multifactor authentication (MFA)? If yes, which applications are critical?
- Is biometric-based passwordless authentication required?
- How often is the MFA required, and what reauthentication triggers need to be deployed?
- Do you have a secure identity database for the full type or contract employees?

Once you have evaluated the responses to the questions, you will then need to devise a policy, plan, and then deploy the solution that allows strong authentication and authorization of the users before they can access any information. User trust can be verified using different methods and actions such as:

- **Multifactor Authentication (MFA):** Ensure users verify their identity through multiple factors (something they know, have, or are)—for instance, a password combined with a mobile authentication app or biometric verification.
- **Role-Based Access Control (RBAC):** Grant users access only based on their roles, ensuring they can access the minimum necessary resources.
- **Contextual Authentication:** Authenticate based on context (time of access, location, device type) to provide a more secure, tailored approach.
- **Just-in-Time Access:** Ensure that users get access only when they need it and that it's revoked immediately after their session ends, minimizing the risk of lingering access.
- **User Behavior Analytics (UBA):** Monitor user activity in real time, including access patterns and application usage. Anomalies (e.g., accessing data from an unusual location or time) trigger additional authentication steps or even block access.
- **Dynamic Risk Scoring:** Assign risk scores to users based on behavior, device posture, and external factors (e.g., suspicious login attempts). Trust levels adjust dynamically based on these scores.

For a greenfield scenario (described more fully later in this chapter), it is easy to plan, design, and implement first. Users are then onboarded to a well-defined and secure environment. But in a brownfield environment, you might face challenges such as users being hesitant to adopt new methods of authentication. Some workflows might need to be interrupted to adopt a new user identity verification system. To overcome some of these challenges, you must do the following:

- Prepare clear communication and messaging for the need for change.
- To avoid any hiccups due to technical glitches, ensure that all systems are tested. You can start the pilot with a small group and data set as identified in the matrix.

Device Trust Definition

Defining device trust means ensuring that a device is safe and authorized to access the network. You must never assume that any device is safe because it is inside the network or is company-owned. Especially with BYOD policies, employees are now allowed to bring their own devices to work. Such devices need to be secured and segmented properly. Ensuring devices accessing your network are in a healthy state is a critical step for zero trust deployment. You can start by doing the following:

- Evaluate the type of the company's own assets, such as model, make, and operating systems.
- Rank devices based on the risk and data sensitivity.

- Create a list of technologies that can be used for device trust such as virtual private network (VPN), mobile device management (MDM), and certificates.
- Create a strategy based on supported devices and their capabilities such that each device can be continuously authenticated.

It is important to understand the difference between a company-owned asset and company-managed asset. A company-owned asset is owned, managed, and fully controlled by the organization, whereas a company-managed-only device typically includes BYOD devices that an employee can purchase but are managed by the company. This is typically done by enrolling devices in an MDM system that defines the security policy and configuration for the device such as certificate requirements or password lengths. You can even wipe an entire device remotely if an employee reports it as stolen. Access to further applications and resources could be constrained for such devices. Because managed devices are tracked, enrolled in configuration and patch management programs, and continuously monitored for security incidents, the trust level for such devices is always higher than for unmanaged personal devices. As such, your zero trust policy usually allows Internet-only access for unmanaged personal devices. We have discussed managed and unmanaged devices, but how do you bring a device under management? You can use a combination of solutions such as

- **Certificate-Based Trust:** In zero trust, every device must prove itself before accessing any resource. Using certificate-based trust, a device proves its identity by showing its digital certificate. Even if the device is already inside the network, it must keep proving its trustworthiness every time it tries to access new resources or data.
- **Device Health:** It is important to ensure that a device is healthy before it can be allowed on the company's network. Even a company-managed device with a certificate installed can get infected. It is important to use device posture as one of the device trust points. Validating if the endpoint security agent is installed to protect against the threats helps in establishing trust.

Device trust further strengthens the security because now an attacker needs not only a valid user credential but also a trusted device to launch any attack. A combination of trusted users with trusted devices is commonly used as part of the zero trust policy definition to provide different levels of access. Typical components you will require for device trust include

- **PKI Infrastructure:** PKI (Public Key Infrastructure) is a framework that manages digital certificates and encryption keys to enable secure communication and authentication over networks. In zero trust PKI is used to issue device certificates. These issued certificates will be used for identifying all company-managed assets, including company-owned or BYOD assets.
- **Security Agents:** A variety of tools can be installed on company-owned assets, such as endpoint agents, VPN, VPNless remote access, and device posture agents. For BYOD, an MDM solution could be used.
- **Central Inventory:** It is important to have a central inventory of all devices in your network with proper asset tags, owners, type of access required, and so on.

One common challenge with device trust is the hesitation of users to install any agent or certificate on their devices due to privacy concerns. This requires careful validation based on the local laws, so the organization must use a solution to verify the trust without violating privacy. As an example, monitoring the website that a user visits on a BYOD could be considered a privacy issue, while suggesting a stronger login password with a specific key length could be considered a genuine safety measure.

With user and device trust policies established, you now know who can access the network using which types of devices. We have covered a high-level approach, but you need to go multiple levels down to establish detailed policies, tools, procedures, and ways to track the user and device trust. Once you have done this, the next step is to enable access to the application by framing policies that a combination of users and devices can access.

Trust Score Calculation

Trust levels can be represented as scores or risk levels based on multiple factors. Here's a simple breakdown of how they might be calculated.

Each factor gets a score, and the total score determines the user/device trust level:

- Base Trust for User Credentials (e.g., MFA or SSO): 50 points
- Device Security (e.g., fully patched, antivirus running): 30 points
- Geolocation (known/unknown location): 10 points
- Time of Access (normal/abnormal): 10 points

If all conditions are met, the trust score would be 100. A lower score (below a set threshold, e.g., 70) might trigger additional security actions, like reauthentication or restricted access.

Instead of adding points, you can subtract trust based on risk indicators. For instance:

- Base Trust: Start at 70.
- Subtract 40 if the user uses only a password (no MFA).
- Subtract 20 if the device is missing recent patches.
- Subtract 10 for unrecognized geolocation.

Once a trust score or risk level is assigned, the system decides how much access to grant and whether any additional actions are needed:

- **High Trust Score (90–100):** Full access to the system, no extra verification needed.
- **Medium Trust Score (70–89):** Limited access, or additional verification required like MFA.
- **Low Trust Score (Below 70):** Deny access or allow access only to less-sensitive resources.

Defining Application Access Policies

The entire idea of establishing trust for users and devices is to allow them access to business applications in a secure way. This ensures that even after the user and device are verified, they get access only to specific applications they need. One of the important considerations you need to make is where the application and data are hosted, such as public or private cloud, on-premises, or SaaS applications. You will need to devise the policies for different data sets and applications based on how often they are accessed, keeping in mind the user experience you want to provide. Following are some of the decision points for creating policy around application access:

- A comprehensive list of applications used in the organization, both internal and external, should include cloud-based, on-premises, and hybrid applications.
- It is also important to identify the application dependencies like interactions with databases, other services, and APIs. This information will come in handy while building the microsegmentation.
- What user types need to access the application? We covered this step from a user authentication perspective earlier in the user trust definition, but now you need to map the application in detail, along with the frequency at which different user groups typically access any applications, and rank the risk of the applications.
- Policies also need to consider device type and location of user groups concerning the application accessed. Do you want to reauthorize or have stricter controls when users try to access the service from a location that is different from their regular location? If your entire workforce is mobile, then your strategy needs to include the fact that random location is normal behavior. In such a case, frequent location-based reauthorization may result in poor user experience. You may want to club the common application sets and allow access based on single sign-on (SSO).
- The SSO-based approach is essential because users often forget passwords for multiple applications. When many applications are involved, users may resort to using the same password for all. Here, SSO provides a centralized enforcement point, allowing the establishment of policies on password length and change frequency. It is advisable to implement multifactor authentication with biometrics alongside SSO.

Access policies should align with user and device trust levels. Trust depends on contextual factors like location, time, and behavior, necessitating continuous monitoring even for authorized access. For example, if a terminated employee quickly downloads multiple files, this could indicate potential malicious intent. Such behavior should trigger alerts or adjust access per established protocols. This strict approach reflects a zero trust framework, which emphasizes that no one is inherently trusted. Ongoing supervision and adaptive policies are vital to enforce zero trust principles.

Macro- and microsegmentation form the basis of the zero trust network access, as you will learn later in this book. From an application perspective, it is also critical that segmentation considers the following constructs:

- **Application-Level Segmentation:** Isolate applications from one another so that a compromise in one doesn't lead to a breach in others. This involves defining granular policies for inter-application communication and blocking unnecessary pathways.
- **Data and API Segmentation:** Segregate databases, APIs, and other backend services from applications and users that don't require direct access.
- **Environment Segmentation:** Separate development, testing, and production environments to ensure that an issue or breach in one environment doesn't spread to others.

It is also important to think about securing the data in motion. When the authorized user groups access specific applications, it must be secure. Following are some of the approaches you may consider:

- **Transport Layer Security (TLS):** Ensure that all traffic between users and applications is encrypted using TLS, preventing interception by attackers.
- **End-to-End Encryption:** For sensitive applications, ensure that data is encrypted at rest and in transit, especially when communicating with external services or across untrusted networks.
- **API Security:** Use API gateways and API security tools to secure communications between applications, especially in cloud environments where APIs are frequently exposed.

Enforcing Policies

With the details gathered so far, now you have a clear understanding of the users, devices, and applications in your environment. It is time to think about how to enforce these policies, monitor the users, and continuously refine them. Because there is a complex matrix of users, devices, and applications, it is important to set a baseline and expand from there. This means a base trust with the users and devices is based on certain factors like

- If the user is seen for the first time
- Which factors are used to authenticate and authorize the users
- If there are signs of malicious activities
- If the device is managed; if yes, company-owned or BYOD?
- Device posture details

If these basic checks pass, you can assign the base trust-level policies that allow access to the common and less risky applications. You can then add further levels of trust

confirmation for the highly sensitive user and application combination. This means that with base trust users can access common applications such as email and Microsoft Office. For example, if the user login happens at the usual time, using the known devices with an attempt to access the regular applications, the decision engine could assign a high trust level with no additional authorization steps required. This will also help you define trust tolerance, as explained later in this chapter.

Granular policies and secondary trust verification are required to access critical assets, including code or financial information. At this point, you also need to decide what actions or changes will result in a loss of trust—for example, a change in hardware. A real-time decision engine is typically used for adaptive access.

Contextual information from the users needs to be collected, stored, and then analyzed to continuously evolve the policies. Advanced AI/ML-based analysis engines could be deployed for this behavior analysis. The information base must have information from events, as described in the following subsections.

Contextual Data

User roles and devices change over time. Employees might change their role for various reasons, such as changing teams or getting a promotion. As a result, the location to access the information also changes. Device type, operating system, and applications used might also change. It is important to collect all this information.

Connection Metadata

It is important to collect user role information and device details at the time of login. User roles could easily be collected from the active directory group memberships, and device details can be collected during posture assessment. Operating system version, installation of mandatory endpoint solutions, device status (e.g., if jailbroken)—all must be collected and stored for further analysis. This also allows you to have a central inventory of all the devices in one place with their security state. This information, when combined with the user information, provides insights into the compliance status of teams and individuals. Necessary actions can be taken if noncompliance is related to a specific group or team.

Logging Suspicious Actions

As you established the baseline trust earlier, it is also important to create the baseline behavior of the users within that trust level. Any deviation from the baseline needs to be recorded and analyzed, and immediate action should be taken. Continuous failed attempts at authorization, use of unsupported platforms to access resources (like jailbroken devices), and attempts to access resources with different user credentials from the same device must all trigger alerts, and immediate action needs to be taken based on company security policies. Additional authorization using MFA, device quarantines, and reduced trust level with each event like this are some of the many actions you can take in these cases. Detailed analysis will help you identify the repeat offenders (intentional or

unintentional). AI-based predictive analysis could be used once sufficient data points are available, to help prevent the attacks, reduce the attack surface, and create a self-healing network. Details on self-healing and maturity levels are covered in the later chapters.

Trust Tolerance

Trust tolerance is the degree of flexibility in the zero trust model. While the fundamental principle is “never trust, always verify,” trust tolerance allows for different levels of stringency depending on various factors like the sensitivity of the resource, user context, or current security posture of the device. It answers the question: “How much risk are we willing to accept for this specific access?”

- **High Sensitivity (Low Trust Tolerance):** When a user is accessing critical systems (e.g., financial records, confidential data), there is minimal tolerance for risk. Trust levels must be high, and any small deviation (like accessing from a new location or using a noncompliant device) should trigger additional security measures or block access.
- **Low Sensitivity (Higher Trust Tolerance):** When a user is accessing less critical resources (e.g., internal HR portals or general information), the system can tolerate more variations in trust. For instance, accessing from an unfamiliar device might still be allowed, but with more monitoring.

In zero trust, you will create a dynamic trust tolerance, which adjusts based on the current risk environment. A user trust value could be very high at the time of initial login and authentication. However, as the user tries to access different applications, the trust level can erode based on that user’s behavior. Every organization needs to devise policies that reflect its trust tolerance. Threshold-based dynamic access policies need to be defined. If the trust level of a user is maintained, the sessions and access levels could be extended by a predefined time. In that case of trust falling below a certain threshold, the user needs to be reauthorized before access to the existing or new applications can be granted.

Several tools can help organizations manage trust tolerance within a zero trust architecture:

- Identity and access management (IAM) tools like Okta or Microsoft Azure Active Directory allow for context-based access policies that adjust trust levels dynamically.
- Security information and event management (SIEM) solutions like Cisco Splunk or Elastic Stack help monitor security events and adjust trust tolerance based on real-time risk factors.
- Endpoint detection and response (EDR) solutions like Cisco Secure Endpoint or CrowdStrike provide continuous monitoring of device security, adjusting trust tolerance based on device posture.

By this point, you must have a clear understanding of the overall approach and key decision points to develop a zero trust policy by defining clear objectives around user and device trust, application access, and policy definition requirements. In the next section, we will explore various tools and technologies that help you achieve these outcomes.

Tools and Technologies

When an organization implements a robust zero trust architecture, a variety of tools and technologies are essential for continuously validating users, devices, and applications at every access point. These solutions work together to ensure that only authenticated and authorized entities are allowed to access specific resources. By leveraging these technologies, organizations can enforce strict security policies, limit potential attack surfaces, and create an environment where trust is never implicitly granted but is verified with each interaction.

Central Inventory

A central user and device inventory is like a list that keeps track of who the users are and what corporate and personal devices they are using. Each user or device has a role or job. These roles tell the system what the user or device is allowed to do. This inventory helps the security system decide if a user or device can access certain information or parts of the network. One of the primary ways to implement zero trust is to assign users and devices to specific domains based on their communication needs. Typically, every organization will maintain the active directory for its users, grouped under different business groups. These active directories are then mapped with authentication servers like the Cisco Identity Services Engine, and different policies can then be created for different sets of users. Additional parameters—device type, posture state, time of day, and so on—can be used in authorization policies.

Having a central repository is one of the prime requirements of zero trust implementation. At the time of writing this chapter, Cisco Identity Services Engine can join up to 50 active directory domains. This allows segmented domains to be brought under a single zero trust policy domain.

Identity and Access Management

Validating the user identity using strong methods is fundamental to the zero trust implementation. Passwords can be easily stolen and reused; as such, strong authentication methods such as MFA need to be part of the user validation. Following are some of the approaches you can take:

- **Multifactor Authentication (MFA):** Ensure users provide two or more ways to prove their identity (for example, a password plus a one-time code sent to their phone).
- **cMFA:** Use tools that support continuous validation of the user identity at fixed intervals or based on triggers.

- **Single Sign-On (SSO):** Allow users to log in once and access multiple applications securely.
- **Role-Based Access Control (RBAC):** Set up permissions based on user roles, ensuring people access only what they need. For example, a marketing team member should not access financial systems.

Tools like Okta, Microsoft Azure Active Directory, or Duo Security can help you manage identities and enforce MFA and SSO.

Network Segmentation

To apply the zero trust polices, a network must be segmented using macro- and microsegmentation approaches. Remember that zero trust is not only about providing secure access; it is also about reducing or limiting the impact of any attack. Network segmentation reduces the attack surface by limiting access to a specific domain.

Macrosegmentation is a way to divide a network into smaller virtual networks (segments), usually based on the types of users, devices, or applications. Each virtual network will have its own security rules. For example, you might create one network segment for employees, another for guests, and a third for sensitive data. In zero trust deployment, macrosegmentation helps by limiting who can access certain parts of the network. Even if someone gets access to one segment, that person can't move freely to other segments without passing additional security checks. This makes it harder for attackers to spread across the network, improving security and reducing the risk of unauthorized access. You will create macrosegmentation using firewall boundaries and virtual segmentation using concepts of virtual routing and forwarding (VRF).

Microsegmentation approaches like VLAN, security group tags (SGTs), or endpoint groups (EPGs) in data centers allow microsegmentation within a macrosegment. The idea is to further restrict and control the communication. With the zero trust principle, only allow communication to what is required. To implement this, smaller network segments are desired. However, the manual assignment of users and devices becomes a management overhead. That is why automated assignment of microsegments is done using AAA servers like ISE.

Figure 3-1 shows high-level macrosegmentation using VRFs and firewalls. You will notice that firewalls are used to isolate the different sections of the network such as the data center, Internet, and BMS/OT areas. The microsegmentation approaches of SGT/VLAN can be used to create microsegmentation within each group. In this example, the enterprise LAN is microsegmented into various segments for corporate laptops, IoT endpoints, and collaboration endpoints. The OT network is using VLAN as a microsegmentation approach with the firewall creating a separate OT zone and macrosegment.

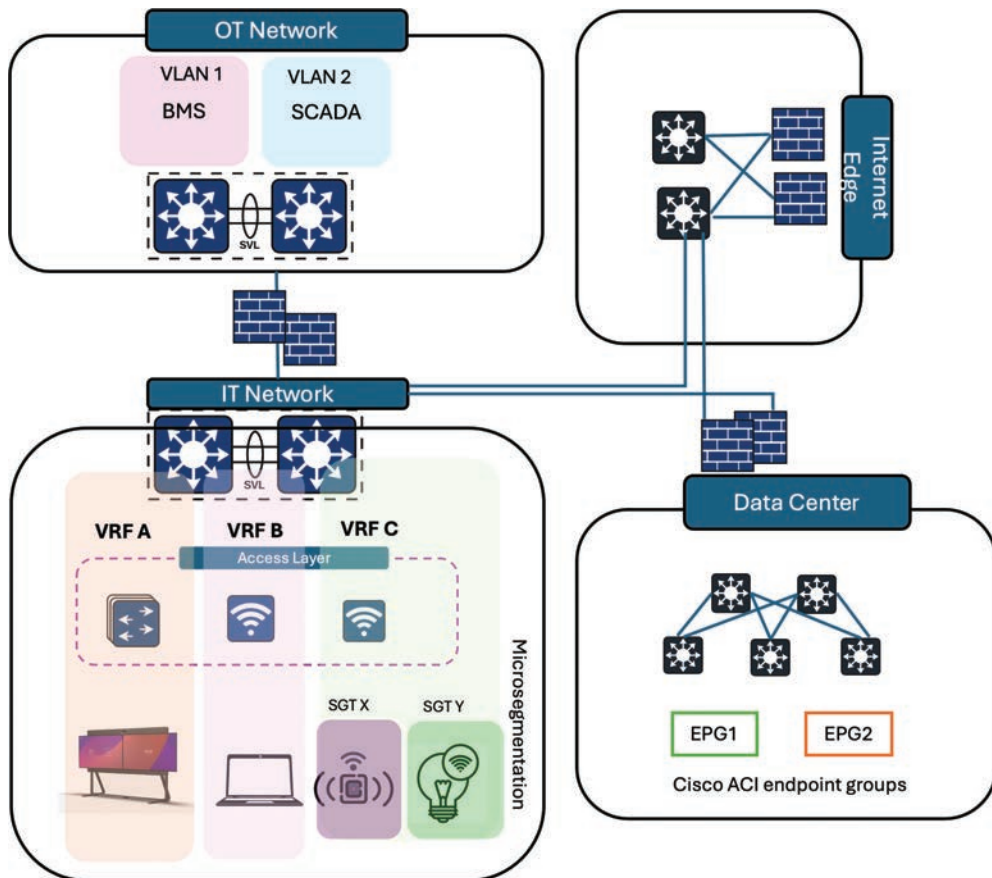


Figure 3-1 Network Segmentation for a Typical Enterprise with IT, OT, and Data Center Blocks

Device Posture with Endpoint Security

Because different types of endpoints will connect to different parts of the network, it is important to ensure these endpoints are healthy. Even though the devices will connect in their own microsegments, still they can pose risks to the network and other devices. Accessing the device health and providing that information to AAA servers allow them to be put in the correct microsegments. If a device is detected as unhealthy, usually it is kept in a quarantine zone with access to only remediation tools.

- **Check Device Health:** Ensure all devices that access your network have security updates and antivirus software installed.
- **Device Authentication:** Verify that devices are authorized to access the network. If a device is not secure (e.g., using outdated software), block or limit its access.
- **Endpoint Detection and Response (EDR):** Monitor devices in real time to detect and respond to potential threats.

Tools like mobile device management (MDM), Jamf, or Microsoft Intune can help control device security and access their posture. Cisco Secure Client provides a modular approach with VPN, Posture, and zero trust network access (ZTNA) modules for zero trust deployment.

Virtual Private Network (VPN)

A virtual private network is a technology that creates a secure connection over the Internet. Please note that not all VPN types offer encryption and authentication. In the context of this section, our focus is on SSL/IPsec remote access VPN technologies. VPN allows users to send and receive data as if their devices were directly connected to a private network, even when they're using public networks like Wi-Fi in a coffee shop or hotel. This secure connection is made possible by encrypting the data being transferred and masking the user's IP address, ensuring privacy and security. In a zero trust environment, users and devices must be verified before accessing sensitive resources. VPNs help enforce this by requiring users to authenticate themselves before they can establish a connection. This authentication process adds an extra layer of protection, ensuring that only trusted users can connect. However, you will not be able to apply granular controls with the VPN as demanded by the zero trust. You will note that many companies are moving away from VPN-based access and adopting VPNless secure access using SASE/SSE. VPN is used only for the legacy use cases and that can also be toward the SSE module in the cloud rather than the data center. What it means is that users will connect via VPN into the SSE module with a VPN concentrator sitting in the cloud. Once the user is connected via VPN, it has to go through the regular security service chain before it can access any kind of data either from cloud service providers like SaaS applications or anything in the company data center. This is explained in detail in the section "Applying Zero Trust Using SSE."

In summary, traditional VPN connections to the organization's data center allow you to tunnel traffic into the organization, but there is no easy way to apply detailed zero trust checks and policies. Doing so is not impossible, but it does make the design complex. At the time of writing this chapter, the industry is moving toward centralized cloud-based ZTN deployments.

Identifying Business Workflows

At the start of this chapter, we presented an approach to identify and define the trust for users and devices to allow access to different applications using zero trust principles. Now at the time of the deployment, you need to convert them into business workflows. Following are some of the common workflows:

- On-premises employee with a trusted device accessing a private application in the local data center
- On-premises employee with a trusted device accessing a private application in the cloud/SaaS
- On-premises contractor with an untrusted device accessing private applications

- On-premises guests with untrusted access accessing the Internet only
- Remote employees with trusted devices accessing private applications in the data center
- Remote employees with trusted devices accessing applications on SaaS

Applying Zero Trust Using SSE

Secure Access Service Edge (SASE) is a cloud-based security model that merges networking and security services into one platform, integrating features like software-defined wide area network (SD-WAN), firewalls, and identity-based access control. Security Service Edge (SSE), a subset of SASE, focuses on security technologies such as Secure Web Gateway (SWG) and Cloud Access Security Broker (CASB). In a zero trust environment, cloud security is crucial as applications move to the cloud. SASE and SSE offer cloud-native tools like CASB to monitor data use and ensure compliance, verifying security continuously even when accessing services remotely. SASE facilitates secure remote connections without traditional VPNs, providing scalable security. SSE safeguards access to cloud apps regardless of user location and, while initially for cloud use, SSE has become the primary method for implementing zero trust due to its simplicity and flexibility. This topic requires detailed discussion for zero trust deployment by any organization in the current era.

Let's look at the components of SSE before we delve into the details of the deployment strategies. SSE has multiple components such as (but not limited to)

- **Firewall as a Service (FWaaS):** FWaaS delivers firewall functionality via a cloud service, enabling organizations to apply security policies throughout their network, even for remote users. It evaluates and controls traffic flowing between users and applications to thwart threat attacks.
- **Secure Web Gateway (SWG):** SWG safeguards users from web-based threats by filtering out unwanted content, blocking malicious websites, and enforcing acceptable use policies. It also inspects encrypted traffic to ensure that threats aren't concealed within SSL/TLS.
- **Cloud Access Security Broker (CASB):** CASB serves as a security gatekeeper between users and cloud services, implementing security policies, tracking usage, and safeguarding sensitive information in cloud applications.
- **Data Loss Prevention (DLP):** DLP is designed to detect and prevent unauthorized access or sharing of sensitive data. It monitors and controls the movement of sensitive information across the network to ensure it is not leaked or accessed by unauthorized users.
- **Secure DNS:** It helps endpoints from rogue websites. User DNS requests are forwarded to the secure DNS module within the SASE platform, where the URLs are validated against the central database for any vulnerability. Only safe websites are resolved for the users, while URLs hosting malicious content are blocked and the user is notified.

Using SSE, you can deploy zero trust for all common use cases including

- Private application access by the users either on-premises or in the cloud/private cloud
- Secure Internet access
- Legacy VPN-based connectivity to resources both on-premises and in the cloud

As identified in the business workflows, different types of users and device combinations will be used in any organization. Based on the user types, devices can be managed or unmanaged. The SSE architecture typically supports client-based access for managed endpoints and clientless access for unmanaged devices. Cisco Secure Access is an example of SSE that provides secure access to the applications on-premises or in the cloud with zero trust components built in.

Next, let's look at the different use-case types and deployment approaches.

Client-Based ZTNA Deployment for Managed Corporate Devices

For this first scenario, a zero trust access module needs to be installed on an endpoint for client-based secure access. Cisco Secure clients have a specific module for zero trust. Other vendors have similar functionality, either as a standalone zero trust architecture (ZTA) module or integrated into other endpoint software solutions. The primary function of the ZTA module is to intercept and send traffic to the SSE in the cloud based on policies defined by network administrators. This works as follows:

- The user tries to open any application on their device. The ZTA client sitting on the laptop controls the traffic routing and usually also handles functions like device posture. Typically, you define which applications need to be routed via SSE and which traffic needs to be sent directly to the Internet. These policies are defined in the SSE module and pushed to the ZTA client on the device.
- This traffic is intercepted and sent to the SSE in the cloud. The method to send this traffic to the SSE client is based on the vendor implementation. Cisco uses the QUIC protocol to send traffic to SSE per application. SSE providers usually have multiple points of presence (PoPs) across the planet. Traffic is usually sent to the nearest POP using anycast IP address.
- Traffic first hits the authentication module that decides whether the user/device combination is allowed to access that specific application. Based on the policies defined by the network admin, further authorization flows such as MFA and device posture checks are triggered.
- SSE also has policies that define how traffic needs to be routed toward its destination. The application may reside in the company's local data center, served from the public cloud, or it can be an SaaS application like Office 365.

- Once the user is authorized, traffic is then routed to the specific destination because SSE usually has direct connections to the specific data (such as IPsec tunnel to the organization's data center, direct high-speed secure connections to cloud service providers).
- The authentication module may trigger periodic reauthorization based on the trust of the device and application access required based on the policies defined in the SSE module.

In this scenario, users can access the required application from anywhere in the world (or from space, as long they have a connection to the SSE portal) without the need for any VPN client. Their application access experience is consistent and without the need to reconnect the company VPN. This method is also known as *VPNless secure access*. Figure 3-2 shows the client-based zero trust access architecture for a corporate device.

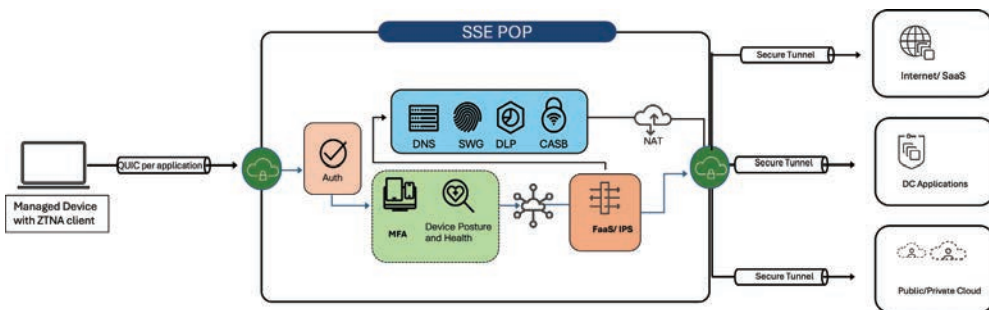


Figure 3-2 ZTNA Client-Based Access Using SSE

Clientless ZTNA Deployment for Unmanaged Devices

In the previous example, because the device was managed, it was easy to install the zero trust access module on the client. However, if the user is a partner or guest and the device is not managed, you cannot use the client-based ZTA access methods. In such cases, you need to rely on the browser-based ZTA access. This is also known as *clientless zero trust access*. It works as follows:

1. The user tries to access the application via the browser.
2. Traffic is sent to the SSE via HTTPS tunnels.
3. Based on the authentication and authorization policies, the user is allowed to access a specific set of applications.
4. Traffic is then sent to its destination either in the public cloud, partner data center, or a company data center.

One important difference in this method is that, because there is no ZTA client present on the device posture, information available to the SSE module is limited and based on the browser data only. In such cases, it is recommended to have more restrictive policies.

Even in this case, there is no need for VPN clients. Figure 3-3 shows the browser-based access architecture to specific applications in the data center/private cloud. Notice that service chains specific to Internet/SaaS applications have been removed from this flow.

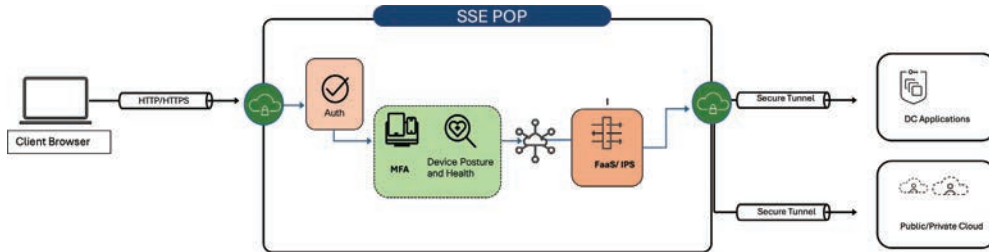


Figure 3-3 *Browser-Based ZTNA*

VPN-Based ZTNA Using SSE

Let's assume that some users require mandatory VPN access. In such cases, you can move your VPN concentrator from the company data center to the SSE cloud. Users will still connect to the SSE via VPN, and from there, security service chains and data policies can remain similar to clientless user access. Figure 3-4 shows the architecture for VPN-based access to private applications only. It is assumed that the Internet/SaaS application could be accessed via split tunneling from the VPN client directly.

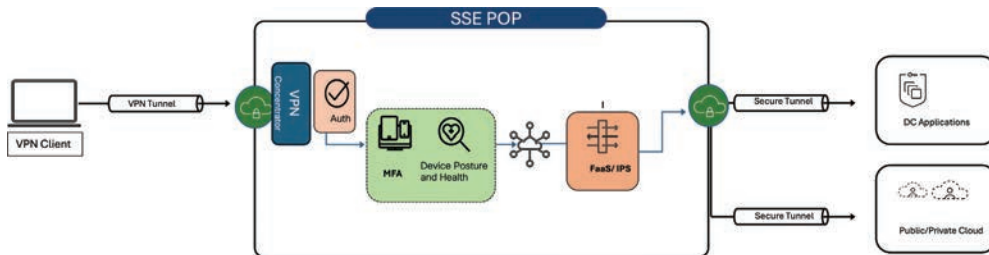


Figure 3-4 *VPN-Based SSE Integration*

SSE Integration for IoT Devices Using SD-WAN

It is not only the users but many devices and IoT devices that need to access their servers via the Internet. You can direct traffic to the SSE in that case also. In such cases, endpoints cannot initiate the tunnels or connections to the SSE POP. Here, technologies like SD-WAN become handy. Assuming that zero trust-based macro- and microsegmentation are already implemented using firewalls, VRFs, and VLANs, traffic from a specific macro- and microsegmentation can then be routed to the SSE using SD-WAN. Most SD-WAN solutions like Cisco SD-WAN can route traffic to a specific destination using a tunneling mechanism based on application type. Cisco SD-WAN calls it *application-aware routing*. You can route traffic toward SaaS applications to SSE and create service

chains specific to SaaS or Internet only. Cisco SD-WAN also supports the auto tunneling capability to Cisco SSE.

Once the traffic hits the SSE, POP traffic can then be passed to Internet/SaaS service providers via security service chains or a NAT module as required. Figure 3-5 shows the users and things traffic via SSE for SaaS application access for users and Internet-only access for things like IoT devices.

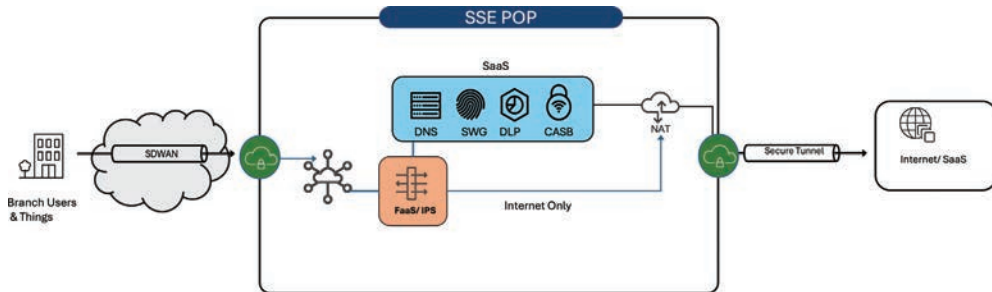


Figure 3-5 SD-WAN-Based SSE Integration

ZTNA Deployment Scenarios

Organizations looking to implement zero trust network access (ZTNA) face different challenges depending on whether they have a greenfield or brownfield environment. Greenfield deployments refer to starting from scratch with no existing infrastructure, allowing for a clean, streamlined implementation of ZTNA principles. In contrast, brownfield deployments involve integrating ZTNA into existing, often complex, infrastructures, which requires careful planning to avoid disrupting current operations. Both approaches come with unique considerations, from resource allocation to compatibility with legacy systems, shaping how zero trust principles are applied. In this section, we will look at the high-level strategy for both greenfield and brownfield scenarios.

Greenfield ZTNA Deployment

In a greenfield ZTN, you can build all infrastructure with a zero trust mindset from the start. This allows for a cleaner and simpler implementation, eliminating the need to manage outdated systems.

In the context of deploying zero trust in a greenfield environment, you can take the following steps based on the strategic steps included earlier:

1. Define the zero trust objective:
 - Formulate a distinct vision for the objectives of zero trust within the organization, such as strengthening security, ensuring better compliance, or gaining improved control over the network access.

- Make sure the zero trust deployment is in sync with overall business goals and strategies. This ensures a strong case for investments and secures executive support.
 - Secure support from senior leadership to guarantee that the initiative receives essential resources and strategic backing import.
- 2. Define a roadmap:**
- Develop a comprehensive roadmap for implementing zero trust. It must outline milestones, timelines, and essential deliverables. Divide the deployment into manageable phases for organized execution implementation.
 - Establish the budget needed for deployment, factoring in expenses for technology, personnel, and training. Distribute resources appropriately to facilitate each phase of the project deployment.
- 3. Develop the architecture and design:**
- Develop a high-level architecture that outlines the application of zero trust principles across the organization. This framework should encompass network segmentation, identity management, and access controls.
 - Design the network layout while considering zero trust principles. Ensure segmentation by establishing zones for various types of assets and data.
 - Plan for microsegmentation to limit lateral movement within the network. Define security boundaries for different workloads and services.
- 4. Deploy:**
- Establish strong IAM systems for overseeing user identities, devices, and applications. Implement multifactor authentication to enhance access security controls.
 - Create granular access policies based on user roles, device types, and the sensitivity of the resources they are accessing.
 - Implement next-generation firewalls, intrusion detection systems/intrusion prevention systems (IDS/IPS), and secure access gateways.
 - Deploy endpoint protection solutions that include antimalware, encryption, and device management.
 - Ensure that applications are securely developed and deployed. Use application firewalls and secure coding practices.
 - Configure access controls to enforce the principle of least privilege. Use tools to continuously evaluate and enforce access policies.
 - Ensure all data in transit and at rest is encrypted. Use strong encryption protocols and key management practices.
 - Set up comprehensive logging and monitoring to track access, detect anomalies, and respond to security incidents.

5. Validate and train:

- Test access policies to ensure they correctly enforce zero trust principles and do not inadvertently allow unauthorized access.
- Develop clear documentation and guidelines for users and administrators on zero trust practices and policies.

6. Adapt and update:

- Regularly update security controls and policies to address emerging threats and changes in the organizational environment.

By following these steps, organizations can effectively deploy zero trust in a greenfield environment, ensuring a secure and adaptive access control framework from the outset.

Brownfield ZTNA Deployment

In a brownfield environment, you deal with established systems, networks, and infrastructure that have been in place for a while. These systems may be outdated and were probably designed without zero trust principles. As a result, there may be existing security vulnerabilities, implicit trust models, or legacy technologies that present challenges for securing them. When you're implementing zero trust in a brownfield, it's essential to thoroughly assess and adjust the current setup while minimizing disruptions to ongoing operations. The aim is to gradually transition to zero trust, identify weaknesses, and bolster security while ensuring smooth operation. Brownfield settings typically require more phased and adaptable strategies to prevent business interruptions, whereas greenfield environments enable quicker and more seamless zero trust integration since they don't require working around existing infrastructure setups.

You can start with the following steps to define your approach to adopt zero trust in brownfield environment.

1. Define objectives and business needs:

- Assess the organization's security goals and what assets are most critical.
- Conduct a comprehensive assessment of the current security landscape, including existing access controls, user privileges, and device management.
- Identify where zero trust is most needed; start with high-risk areas such as critical data or sensitive applications.
- Define the scope of the zero trust implementation, identifying critical applications, sensitive data, and high-risk areas that require immediate attention.

2. Outline the current infrastructure:

- Conduct a thorough audit of the current network, devices, and user access points.
- Identify gaps where security is lacking or where implicit trust exists that needs to be eliminated.
- Inventory all applications, devices, and users to understand the current access state and any potential issues vulnerabilities.

3. Establish user and device trust:
 - Start by establishing a baseline level of trust for all users and devices, regardless of their current access privileges.
 - Deploy multifactor authentication to strengthen user identity verification.
 - Implement device posture checks to ensure that all devices are secured and compliant before access is granted.
4. Segment the network:
 - Assess whether legacy applications can be segmented or modernized to reduce security risks and enhance overall security posture within the brownfield environment.
 - Apply microsegmentation to limit access to sensitive resources. Each user or device gets access only to the resources they specifically need.
5. Implement adaptive policies:
 - Develop dynamic policies that can adapt based on real-time user behavior, device status, or location.
 - Implement measures to enhance device visibility within the brownfield environment to improve security and reduce risks associated with legacy technologies.
6. Begin with small steps, then scale:
 - Start by applying zero trust principles to a single department or system.
 - Test the framework, gather feedback, and expand gradually across the organization.
7. Continuously monitor and improve:
 - Use security monitoring tools like SIEM to detect anomalies.
 - Regularly review and adjust access controls, policies, and system configurations.
 - Continuously iterate on the zero trust implementation, incorporating feedback, insights, and lessons learned to improve security posture over time.

Summary

In this chapter, you learned about the approach for zero trust deployment by first defining the strategy for user and device trust, and application access policies, and then you learned methods for deploying those policies. You also learned about common tools and technologies, including the SSE approach to adopting zero trust. Lastly, you looked at greenfield and brownfield approaches to zero trust deployment.

Index

Numerics

4C's of Cloud-Native Security

cloud, 453

*challenges in standardizing
IAM policies, 453–454*

*encryption challenges and
solutions, 454*

clusters, 454–456

code, 457–458

containers, 456–457

7 R's model, 583–586

802.1x, 240–243

A

**AAA (authentication, authorization,
and accounting), 220. *See also*
authentication; authorization**

access/access control, 619–620

in distributed systems, 444

JIT (just-in-time), 6

just-enough, 6

least-privilege, 12, 17, 31–32

policies, 59–60

role-based, 6

smart contract-based, 523–530

ACLs (access control lists), 404–405

Adaptive Policy, 355, 369, 511–513.

See also policy/ies

**ADM (Application Dependency
Mapping), 625–626, 627**

agent, PnP, 151

AI/ML, 33–34, 48, 426

for adaptive policy enforcement
in hybrid cloud environments,
511–513

anticipating future threats, 426–427

automation, 47

generative, 50

reinforcement learning, 48

role in enhancing cloud-native
security, 425–426

role in policy management, 565–566

supervised learning, 48

unsupervised learning, 48

alarms, Meraki, 345

**ALB (application load balancer),
406–407**

- features and use cases, 407–409
- OSS (open-source software),
 - Kubernetes integration, 409–411
- algorithms**
 - asymmetric encryption, 266–268
 - Grover's, 269
 - hashing, 303
 - RSA (Rivest-Shamir-Adleman), 267
 - Shor's, 268–269
 - symmetric encryption, 266
- analytics, 9**
- anchoring, 351–354**
- anomaly detection, 18**
- Ansible, 510, 766**
 - claiming devices, 177
 - nodes, 44–45
 - playbooks, 44–45
- anycast address, 119**
- AP (access point), salt-and-pepper placement, 344. *See also* wireless networks/Wi-Fi**
- Apache Mesos, 387**
- API/s, 34–35, 359, 439**
 - authentication, 36, 233
 - authorization, 36–37
 - automation, 35–36
 - based communication, 390
 - based LAN automation provisioning, 143–144
 - DAST (dynamic application security testing), 37–38
 - declarative, 391–392
 - eastbound, 35
 - gateway, 399, 468, 473
 - northbound, 35
 - rate limiting, 37
 - resource limiting, 469–470
 - SDK (software development kit), 45
 - security, 36–37, 459, 467–470
 - segmentation, 60
 - southbound, 35
 - westbound, 36
- APM (Application Performance Management), 626, 627. *See also* application/s**
- application/s**
 - access policies, 59–60
 - aware routing, 70–71
 - compatibility issues, 603–604
 - level segmentation, 60
 - modernizing, 392
 - security. *See* cloud-native, application security
- APTs (advanced persistent threats), 4**
- Aqua Security, 388**
- architectures**
 - Cisco Cyber Vision, 699
 - cloud-native, 384–386, 439
 - agility and scalability, 389*
 - API-based communication, 390*
 - containerization, 386*
 - core principles, 391–392*
 - dynamic orchestration and management, 387–388*
 - immutable infrastructure and scalability, 389*
 - integrating DevOps and DevSecOps, 388–389*
 - microservices, 387*
 - observability, 390*
 - resiliency, 389–390*
 - CPwE (Converged Plantwide Ethernet), 695–696
 - deploying
 - IaC tools, 766–769*

- network controllers vs. direct to device, 765–766*
- serverless, 470
 - attack vectors, 475–476*
 - key security challenges, 471–472*
 - security best practices, 472–475*
 - security flow, 477–482*
 - security-first approach, 475*
 - shared responsibility model, 471*
- areas, OSPF, 193
- assessment
 - security, 397
 - tools, 616–618
- asset management, 9
- assume breach mindset, 7
- asymmetric cryptography, 231
- asymmetric encryption, 266–268
- Attack Surface Management. *See* Cisco ASM
- attack/s
 - DDoS, 397
 - DHCP starvation, 112
 - injection, 444
 - MAC address spoofing, 237–238
 - man-in-the-middle, 112–113
 - on-path, 236–237, 239
 - reconnaissance, 39
 - rogue DHCP server, 109–111
 - vectors in serverless architectures, 475–476
- audit/audit trail
 - firewall, 81–82
 - logging, 532–534
 - reputability, 687–688
- resilience, 686–687
- authentication, 6, 220, 223. *See also* MAC authentication bypass; network/s, access control
 - 802.1x, 240–243
 - API, 36
 - centralized server-based, 225–228
 - DHCP, 114–115
 - DID (decentralized identity), 523–530
 - in distributed systems, 444
 - local, 224–225
 - multifactor, 5–6, 18, 31, 234–235
 - PAP (Password Authentication Protocol), 223–224
 - resilience, 681–684
 - REST-API, 233
 - service accounts, 228–231
 - x.509 certificate-based, 231–233
- authorization
 - API, 36–37
 - change of, 245
 - Kubernetes, 519–520
 - module, 390
 - monitoring, 248–249
 - PnP devices, 163–164
 - RADIUS/TACACS+, 243–245
 - resilience, 681–684
 - smart contract, 526–527
 - time-based, 250
- auto-install, 150
- automated analysis solutions, 663–664
- automation, 9, 20, 29–31, 128. *See also* LAN automation; orchestration; SOAR (security orchestration, automation, and response)

AI, 47

APIs, 35–37

device deployment and management, 746–748, 756–757

- candidate configuration datastore*, 752–753
- Python script*, 751–752
- RESTCONF*, 749–750
- TCL script*, 748–749
- using IP SLA and RESTCONF*, 753–756

Fast Burger, 376–377

IaC (Infrastructure as Code), 41

- data modeling*, 41–42
- data serialization formats*, 42–43

infrastructure, 446

large-scale SDN deployment, 369–370

ML (machine language), 48

- reinforcement learning*, 48
- supervised learning*, 48
- unsupervised learning*, 48

network

- evolution of*, 32–34
- maturity*, 34–35

partial automated deployment, 144–147

pre-validation, 775–776

regulatory requirements and compliance monitoring, 424

role in migration, 590

script-based, 33

security, 405–406

segmentation policy, 510–513

for threat containment, 705–706

workflow, 358–359

availability, metrics, 668

AWS. *See also* hyperscaler cloud environments

- application load balancers, 406
- external access to third parties, 657–658
- IAM (identity and access management), 661–663
- Migration Hub, 590
- Well-Architected Framework, 587

AZs (availability zones), 572

Azure, external access to third parties, 658–660

Azure Migrate, 590

B

bandwidth

- planning, 216
- secure data transfer, 599–600

base trust, 60–61

BB84 protocol, 271–272

Bell state, 263–264

best practices

- IAM implementation, 653
- Layer 2 design, 340–342
- Layer 3 design, 343
- macrosegmentation, 85–88
- microsegmentation, 85–88
- network design, 338–339
- VN design, 640–641

BeyondCorp initiative, 2

BFD (bidirectional forwarding detection), 285–286

BGP, 190

BGP-EVPN, 201–202

- control plane, 202–203
- data plane, 203

- management plane, 204
 - policy plane, 203–204
 - blockchain**
 - based storage, 687–688
 - cost and scalability, 522–523
 - immutability, 517–518
 - latency concerns, 521–522
 - BOOTP (Bootstrap Protocol), 104**
 - Bra-ket notation, 256–257**
 - breach/es, 4–6, 16**
 - assumption of, 7, 32
 - containment, 7
 - brownfield**
 - user trust, 56
 - ZTNA deployment, 73–74
 - business workflows, 66–67**
 - BYOD policy, 56–57, 58**
- ## C
-
- cabling, tiered network design, 340**
 - campus network, 127–128**
 - clean-slate approach, 128–129
 - fabric-based. *See* fabric-based network
 - firewall connectivity, 728–734
 - IP addressing, 129–130
 - maintaining continuity, 132–134*
 - management IP addresses, 131*
 - overlay, 131–132*
 - underlay infrastructure, 130–131*
 - LAN automation, 136–137
 - API-based, 143–144*
 - design stage, 139*
 - discovery stage, 139–140*
 - planning stage, 138–139*
 - process and workflow, 137–138*
 - provisioning stage, 140–143*
 - partial automated deployment, 144–147
 - planning, 128–129
 - security considerations, 727
 - site hierarchy, 135
 - upgrades, 128
 - candidate configuration datastore, 752–753**
 - CapEx, 572**
 - CARTA (Continuous Adaptive Risk and Trust Assessment) model, 12**
 - CASB (Cloud Access Security Broker), 67, 539–540**
 - CBAR (controller-based application recognition), 246**
 - cd network-config command, 371**
 - central inventory, 63**
 - centralized server-based authentication, 225–228**
 - certificate/s**
 - based trust, 57
 - PnP (Plug-and-Play), 151–154
 - x.509, 231–233
 - certification**
 - Cloud Security Alliance, 25
 - CMMC, 25–26
 - choosing a cloud service model, 579**
 - FaaS (Function as a Service), 581
 - PaaS (Platform as a Service), 580
 - SaaS (Software as a Service), 580
 - CI/CD pipeline, 388–389, 767–768**
 - integrating policy checks into, 564
 - integrating security, 448
 - Meraki as Code, 373–376

- CI/CD/CT (Continuous Integration/Continuous Development/Continuous Testing), 41, 46–47
- circuits, quantum, 264
- CIS (Center for Internet Security), 221, 394
- CISA (Cybersecurity and Infrastructure Security Agency) Zero Trust Maturity Model, 13, 432
- Cisco ACI (Application Centric Infrastructure), 207–208, 405, 497
 - contracts, 501
 - filters*, 502–503
 - for network management*, 503–505
 - provider-consumer model*, 502
 - subjects*, 502–503
 - control plane, 208
 - data plane, 208
 - management plane, 209
 - policy model, 498
 - policy plane, 209
- Cisco Annual Internet Report, 185–186
- Cisco ASM, 416–417
- Cisco Catalyst Center, 85–86, 134, 143
 - LAN automation, 136–137
 - API-based*, 143–144
 - design stage*, 139
 - discovery stage*, 139–140
 - planning stage*, 138–139
 - process and workflow*, 137–138
 - provisioning stage*, 140–143
 - monitoring wireless networks, 345
 - partial automated deployment, 144–147
 - PnP onboarding process, 151–152.
 - See also* PnP (Plug-and-Play)
 - site hierarchy, 135
 - templates, 169–170
 - composite*, 170
 - standard*, 169–170
 - web certificate, 152–154
 - ZTP
 - claiming devices*, 166–168
 - software and hardware deployment selection*, 166–167
- Cisco Cloud Controller, 497
- Cisco Cloud Network Controller, 495
- Cisco Cyber Vision, 699–702, 705–706
- Cisco DUO, 234–235
- Cisco Identity Intelligence, 222–223
- Cisco Identity Services Engine
 - dashboards, 227
 - Log Analytics, 228
 - NAT anomaly detection, 246–248
 - System 360, 228
- Cisco Meraki, 358. *See also* Meraki as Code
 - Adaptive Policy, 355
 - Air Marshal, 345–346
 - alarms, 345
 - claiming devices, 168–169
 - dashboards, 286–287
 - monitoring RF environment*, 347–349
 - spread spectrogram*, 347
 - MX devices, 367
 - onboarding a PnP device, 164–165
- Cisco Prime Infrastructure, 150
- Cisco SAFE security reference model, 434–435

- Cisco Secure, cloud-native tools, 419–421
- Cisco Secure Workload, 418–419
- Cisco Security Cloud, 566
 - Cisco Breach Protection Suite, 568
 - Cisco Cloud Protection Suite, 567
 - Cisco User Protection Suite, 566–567
- Cisco ThousandEyes, 635, 636
- Cisco TrustSec, 85–86, 206, 298
 - configuration policies, 83–85
 - control plane transport, 96
 - LISP, 96–97
 - static, 97
 - SXP, 96
 - enforcement points, 356
 - GRE encapsulation, 94
 - inline tagging, 92–94
 - methods of transport, 91–92
 - monitoring, 354–356
 - SGTs (security group tags), 83
- Cisco XDR, 705–706
- Cisco Zero Trust Framework, 432–433
- claiming devices
 - in Catalyst Center, 166–168
 - in the Meraki dashboard, 168–169
 - using Ansible, 177
 - using direct API calls, 172–177
- classical computers, 254
 - bit, 255
 - gate operations, 259–260
- classless protocol, 194
- clean-slate approach, 128–129
- client and server resilience, 684–686
- client-based ZTNA deployment, 68–69
- clientless ZTNA deployment, 69–70
- cloud environments. *See also*
 - hyperscaler cloud environments
 - adjusting security controls, 646
 - AZs (availability zones), 572
 - choosing a provider, 644
 - choosing a service model, 579, 644
 - FaaS (Function as a Service), 581
 - hybrid. *See* hybrid cloud
 - IaaS (Infrastructure as a Service), 579–580
 - KMS (key management systems), 400–402
 - PaaS (Platform as a Service), 580
 - regions, 572
 - SaaS (Software as a Service), 580
 - shared responsibility model, 393–394
 - SSE (Security Service Edge), 67–68
 - client-based ZTNA deployment, 68–69
 - clientless ZTNA deployment, 69–70
 - SD-WAN-based integration, 70–71
 - VPN-based integration, 70
 - universal best practices, 639–640
 - zero trust
 - continuous monitoring and threat detection, 22
 - encryption and data protection, 22
 - explicit verification, 20–21
 - incident response, 23
 - least-privilege access, 21
 - policy enforcement, 23
 - recommendations, 19–20
 - segmentation, 21

- SOAR (security orchestration, automation, and response), 22*
- cloud policy models, 499–500
- Cloud Security Alliance, certification, 25
- Cloud Smart, 578
- cloud-native, 482–485. *See also*
 - CNAPP (cloud-native application protection platform) solutions
 - application security, 383
 - definition and scope, 438*
 - evolution from traditional security, 439*
 - key challenges, 438–439*
 - architecture, 384–386
 - agility and scalability, 389*
 - API-based communication, 390*
 - containerization, 386*
 - core principles, 391–392*
 - dynamic orchestration and management, 387–388*
 - immutable infrastructure and scalability, 389*
 - integrating DevOps and DevSecOps, 388–389*
 - microservices, 387*
 - observability, 390*
 - resiliency, 389–390*
 - CTEM (continuous threat exposure management), 427–430
 - DID (decentralized identity), 518–520
 - emerging trends, 482–483
 - HashiCorp Vault, 402–404
 - incorporating matured zero-trust frameworks, 431–433
 - leveraging OpenTelemetry for security, 422–423
 - monitoring and logging, 421–422
 - observability, 627
 - regulatory requirements and compliance monitoring, 424
 - role of AI and ML in enhancing security postures, 425–426
 - security, 390–391, 411–413, 423, 445. *See also* 4C's of Cloud-Native Security
 - app definition and development, 447–448*
 - authentication and access control in distributed systems, 444*
 - Cisco Secure technologies, 419–421*
 - CNCF projects, 450–453*
 - injection and misconfiguration risks, 444*
 - logging and monitoring, 445*
 - managing component vulnerabilities as scale, 444–445*
 - microservices, 392*
 - observability and analysis, 448*
 - orchestration and management, 447*
 - platforms, 448–450*
 - provisioning, 446*
 - runtime, 446–447*
 - segmentation, 496–497
 - service-to-service communication, 392–393
 - standards, 430–431
 - visibility and transparency across the stack, 422
- CMMC (Cybersecurity Maturity Model Certification), 25–26

CNAPP (cloud-native application protection platform) solutions, 458–459, 482–485

API security, 459

building secure applications with cloud-native security, 460

managing dynamic cloud configurations, 464

API security, 467–468

automated remediation tools, 464–465

configuration management tools, 464

continuous monitoring and drift detection, 464

dynamic secrets management, 464

IaC and security, 465

secure APIs using Kubernetes security, 468–470

securing the software supply chain, 465–467

policy enforcement, 459

runtime protection, 459

security in application design and development, 460

immutable infrastructure, 461

principle of least privilege, 460–461

secrets management, 461

secure coding practices, 462

shift-left security and DevSecOps integration, 463–464

vulnerability management, 459

CNCF (Cloud Native Computing Foundation), 430–431

projects for cloud-native security, 445, 450–453

app definition and development, 447–448

observability and analysis, 448

orchestration and management, 447

platforms, 448–450

provisioning, 446

runtime, 446–447

CNOT gate, 262–264

CoA (change of authorization), 245

“as code” provisioning, 772–775

code/coding

infrastructure as. *See* IaC (Infrastructure as Code)

input validation, 457

secure practices, 462

DAST (dynamic application security testing), 462

SAST (static application security testing), 462

smart contracts, 520–521

co-location, 577–578

commands

IOS

cd network-config, 371

debug ip routing, 283

hw-module beacon slot 1, 178

ip cef load-sharing algorithm, 303

no port-channel standalone-disable, 162

show bfd neighbors detail, 286

show cts environment-data, 88–89

show cts role-based counters, 313

show cts role-based stg-map all, 89–90

- show cts role-based stg-map vrf*, 97, 308–309
- show cts sxp connections brief*, 311–312
- show ip cef exact-route*, 181–182
- show ip cef vrf ENT exec-route*, 301
- show ip dhcp snooping*, 111
- show ip route super*, 282, 283
- show ipv6 interface*, 118
- show lisp instance-id*, 302
- show netconf-yang ssh server command*, 770
- show run*, 97, 759–761
- show run all*, 238
- iperf, 290–296
- Common Policy**, 538–539, 735–736
- communication**
 - API-based, 390
 - service-to-service, 392–393
- compliance**
 - automated, 398
 - cloud-native security, 421–422
 - continuous, 564
 - enforcement, 446
 - network security, 246–248
 - regulatory, 424, 460, 543, 575–576, 607
 - reporting, 563
 - SDN environment security, 334–335
 - segmentation and, 492
- composite template**, 170
- computing**, need for new technologies, 254. *See also* quantum computing
- conduit**, 698
- confidential computing**, 447
- confidentiality**, 594
- configuration file**, minimum base underlay, 145–146
- configuration management**, 371–373
- configuration policies**, 83–85
- congestion**, 216
- container/s**, 386
 - isolation, 469
 - orchestration, 387–388
- containment**, 346
- continuity**, IP addressing, 132–134
- contracts**, 501
 - filters, 502–503
 - network management, 503–505
 - provider-consumer model, 502
 - smart, 517, 520–521
 - for policy enforcement*, 530–532
 - revocation challenges*, 522
 - subjects, 502–503
- control plane**
 - ACI, 208
 - BGP-EVPN, 202–203
 - MPLS-VPN, 200
 - SD-Access, 205–206
 - SD-WAN, 210
- convergence**, 279–280
 - in data center networks, 290–296
 - in Layer 3 routed architectures, 281–284
 - monitoring, 308–314
 - routing protocol dependency, 280–281
 - server-side verification, 288
 - traffic captures*, 289
 - Wireshark filters*, 289–290
 - testing, 300–303

- stateful traffic generation*, 304–305
- stateless traffic generation*, 305–307
- traffic generation*, 303–304
- timers, 284–288
- COOP (Council of Oracles Protocol), 208
- core principles of zero trust
 - assume breach, 7, 32
 - explicit verification, 5–6, 19, 31
 - least-privilege access, 6, 19, 21, 31–32
- correlation, for threat containment, 705–706
- cost, OSPF, 193
- COTS (commercial off-the-shelf) software, 675–676
- COVID-19 pandemic, 11, 680
- CPwE (Converged Plantwide Ethernet) architecture, 695–696
- credentials, local, 224–225
- CRL (certificate revocation list), 232
- Crossplane, 510
- CRQC (cryptanalytically relevant quantum computer), 269
- cryptography
 - KMS in the cloud, 400–402
 - post-quantum, 265–266, 270
- CSA (Cloud Security Alliance) Zero Trust Working Group, 12–13
- CSPM (cloud security posture management), 415
 - benefits, 415
 - features, 415
 - future of, 415–417
 - relationship to zero trust, 419
- CSRF (cross-site forgery) attack, 38

- CTEM (continuous threat exposure management), 427–430
- CVDs (Cisco Validated Designs), 338
- CV-QKD (Continuous Variable QKD), 271
- CWPP (cloud workload protection platforms), 417–418
- cybersecurity, 4, 382. *See also* security

D

- DAI (Dynamic ARP Inspection), 112–113
- dashboards, 227
 - Cisco Catalyst Center, 345–346
 - Cisco Cyber Vision, 701
 - Cisco Meraki, 286–287, 345–346, 347
- DAST (dynamic application security testing), 37–38, 462
- data center segmentation, 571
- data collection, user
 - connection metadata, 61
 - contextual data, 61
 - logging suspicious actions, 61–62
- data in motion, 60
- data lake, 50–51
- data migration, 597
 - Direct Connect Services, 596
 - hybrid, 597
 - network, 595–596
 - offline, 597
 - online, 595
- data modeling, 41–42, 764–765
- data plane
 - ACI, 208
 - BGP-EVPN, 203

- MPLS-VPN, 200–201
- SD-Access, 206
- SD-WAN, 210–211
- data protection policy, 543
- data security, 593
- data serialization formats, 42–43
- data transformation, 602
- DDoS (distributed denial-of-service) attacks, 397
- DE.AE-3 framework, 221–222
- debug ip routing command, 283
- declarative APIs, 391–392
- decoherence time, 258
- decommissioning, 609
- deepfake, 16
- dependency/ies
 - application, 625–626
 - tracking, 466
- de-perimeterization, 2
- deployment and migration phases, SDN lifecycle, 324
 - deploying the SDN controller
 - installing and configuring the SDN controller, 326–327*
 - prioritizing network services for SDN migration, 328–329*
 - setting up high availability and disaster recovery, 327–328*
 - testing and validating migration success, 329*
 - preparing the network infrastructure, 325–326
- design stage, LAN automation, 139
- device/s. *See also* PnP (Plug-and-Play)
 - auto-install, 150
 - automating deployment and management activities, 746–748, 756–757
 - candidate configuration datastore, 752–753*
 - Python script, 751–752*
 - RESTCONF, 749–750*
 - TCL script, 748–749*
 - using IP SLA and RESTCONF, 753–756*
 - claiming
 - in Catalyst Center, 166–168*
 - in the Meraki dashboard, 168–169*
 - using Ansible, 177*
 - using direct API calls, 172–177*
 - decommissioning, 609
 - fully compliant, 137
 - health, 57
 - inventory, 63
 - IoT, 692
 - smart, 32
 - trust, 56–58
 - ZTP (zero-touch provisioning), 136–137
- DevOps, 40–41
 - CI/CD pipeline, 388–389, 448
 - IaC (Infrastructure as Code), 405–406. *See also* IaC (Infrastructure as Code)
 - top cloud security risks in, 463
- DevSecOps, 463–464, 474
- DHCP (Dynamic Host Configuration Protocol), 105
 - authentication, 114–115
 - discover message, 106
 - frame exchange, 105–106
 - man-in-the-middle attack, 112–113
 - Option 6, 157
 - Option 15, 157
 - Option 43, 157, 159–160

- Option 60, 157–159
 - Option 82, 107–109
 - Option 90, 115
 - options, 113
 - PnP (Plug-and-Play), 156–157
 - relay agent, 106
 - rogue servers, 109–111
 - security, 107
 - starvation, 112
 - stateful, 120–122
 - DHCPv6, options, 122**
 - DHCPv6 Guard, 123–124**
 - DIA (Direct Internet Access), 214**
 - DID (decentralized identity), 515–516**
 - authentication, 523–530
 - in cloud-native and hybrid environments, 518–520
 - latency concerns, 521–522
 - Diffie–Hellman key exchange protocols, 266–267**
 - digital identity, 220**
 - Direct Connect Services, 596**
 - DISA Zero Trust Framework, 431–432**
 - discovery stage, LAN automation, 139–140**
 - DiVincenzo criteria for quantum computing, 258**
 - DLP (data loss prevention), 9, 67**
 - DNS (Domain Name System)**
 - best practices for cloud migrations, 642–643
 - PNP, 160–161
 - Docker, 386**
 - domain resilience, 672–673**
 - DoS (denial-of-service) attack, DHCP starvation, 112**
 - downtime, 573, 601**
 - DR (disaster recovery), 327–328, 643, 645**
 - drift detection tools, 464**
 - dynamic addressing**
 - BOOTP, 104
 - DHCP (Dynamic Host Configuration Protocol), 105
 - authentication, 114–115*
 - discover message, 106*
 - frame exchange, 105–106*
 - man-in-the-middle attack, 112–113*
 - Option 82, 107–109*
 - options, 113*
 - relay agent, 106*
 - rogue servers, 109–111*
 - security, 107*
 - starvation, 112*
 - RARP (Reverse Address Resolution Protocol), 104
 - zero trust approach, 109
 - dynamic playbook, 39–40**
- ## E
-
- E91 protocol, 271**
 - EAP (Extensible Authentication Protocol), 240–243**
 - eastbound API, 35**
 - ECMP (equal-cost multipath), 196**
 - EDR (endpoint detection and response), 62**
 - EIGRP (Enhanced Interior Gateway Routing Protocol), 190–191**
 - redistribution, 191–192
 - routing policies, 192
 - scale considerations, 192

EN (Extended Node), SD-Access,
711–712

encryption, 9, 20, 22, 265, 454

API communication, 467

asymmetric, 266–268

in hyperscaler cloud environments,
395–396

quantum-resistant, 270

as a service, 403

symmetric, 266, 271–272

endpoint security, 8–9, 20, 23, 65–66

end-to-end segmentation, 493, 718,
719–723

enforcement. *See* policy, enforcement

enforcement points, Cisco TrustSec,
356

environment segmentation, 60

environmental redundancy, 672

evolving threat landscape, 3–4

EVPN, 300

Expect, 746–747

explicit verification, 5–6, 19, 20–21,
31

F

FaaS (Function as a Service), 581

fabric-based network, 128. *See also*
overlay; underlay

IP addressing

maintaining continuity,
132–134

management IP addresses, 131

overlay, 131–132

underlay infrastructure,
130–131

LAN automation, 136–137

API-based, 143–144

design stage, 139

discovery stage, 139–140

partial automated deployment,
144–147

planning stage, 138–139

process and workflow, 137–138

provisioning stage, 140–143

failure domain, 82

Falco, 456–457

Fast Burger SDN deployment,
377–379

automation, 369–370, 376–377

CI/CD pipeline, 373–376

Cisco Secure Connect, 365–366

configuration management, 371–373

dynamic VLAN assignment,
368–369

ISE integration, 368

Layer 2, 364–365

Layer 3, 365

Meraki as Code, 370

MX security features, 367–368

network design, 362–363

ordering and delivery, 366–367

physical hardware, 363–364

security configuration, 369

using Git for configuration manage-
ment, 371–373

FIAB (fabric in a box), 182–183

filters, Cisco ACI, 502–503

FIPS (Federal Information Processing
Standards), 24–25

firewall

audit trail, 81–82

connectivity in the campus, 728–734

industrial networking, 696–698

third-party integrations, 735–743

traditional, 392–393

First Hop Security, 123

- DHCPv6 Guard, 123–124
- IPv6 Destination Guard, 124–125
- ND Suppress Multicast, 126
- Prefix Guard, 125
- RA throttling, 125–126
- rogue RA, 123
- Source Guard, 125

Flexera’s 2024 State of the Cloud Report, 573, 581–582

Forrester’s Zero Trust eXtended (ZTX) framework, 12

FQDN (fully qualified domain name), 151, 153, 160–161

framework/s

- AWS Well-Architected, 587
- Cisco SAFE security reference model, 434–435
- Common Policy, 538–539, 735–736
- policy, 492, 537
- SAMM (Software Assurance Maturity Model). *See* SAMM (Software Assurance Maturity Model)
- software development, 553–557
- zero trust, 431–434

functional pillars, zero trust, 53–54

FWaaS (Firewall as a Service), 67

G

gate operations

- classical, 259–260
- quantum, 260–261
 - CNOT gate, 262–264*
 - H gate, 261–262*

GCP, external access to third parties, 660–661

GenAI, 50

geo-routing, 218

GIR (graceful insertion and removal), 290–291

Git, 371–373

Google

- BeyondCorp initiative, 2
- Cloud Migration Center, 590

Google Cloud Platform. *See* hyperscaler cloud environments

governance, 9–10, 559–561

- aligning with security policies, 543
- automated, 398

GPS (Global Positioning System), 187

Grafana, 512

GRE encapsulation, 94

greenfield, 745–746

- user trust, 56
- ZTNA deployment, 71–73

Grover’s algorithm, 269

gRPC, 387

GUA (Global Unicast Address), 116

H

H gate, 261–262

HA (high availability)

- cloud migration, 643
- SDN controller, 327–328

hairpinning, 80

HashiCorp Vault, 402–404

hashing algorithm, 303

HCL (Hashicorp Configuration Language), 46, 766–767

HIPAA (Health Insurance Portability and Accountability Act), 26

Humby, Clive, 670

hw-module beacon slot 1 command, 178

hybrid cloud

AI/ML for adaptive policy enforcement, 511–513

Cisco ASM. *See* Cisco ASM

DID (decentralized identity), 518–520

IAM (identity and access management), 649–653

policy enforcement, 565

security, 413–414

segmentation, 491, 506–509

addressing, 493

dynamic workloads, 492–493

inconsistent policy frameworks, 492

visibility and compliance, 492

visibility, 513

zero trust, 23–24

hybrid data migration, 597

hybrid IT infrastructure, 576

Cloud Smart, 578

co-location, 577–578

multicloud strategy, 576–577

on-premises infrastructure, 576

hyperscaler cloud environments, 394–395. *See also* AWS; Azure

automated analysis solutions, 663–664

automated compliance and governance, 398

data encryption, 395–396

IAM (identity and access management), 395

network and infrastructure security, 396–397

threat detection and response, 398–400

unified security models, 395

Hystrix, 390

IaaS (Infrastructure as a Service), 579–580

IaC (Infrastructure as Code), 33, 172, 358–359, 396, 405–406, 465, 510

“as code” provisioning, 772–775

data modeling, 41–42, 764–765

data serialization formats, 42–43

deploying an architecture

network controllers vs. direct to device, 765–766

tools, 766–769

Meraki as Code, 370

network management protocols, 43–44

Terraform, 46

IAM (identity and access management), 8, 63–64, 382, 395

in hybrid environments, 649–653

multicloud, 542–543

permissions, 654–656

policies, 541

managing privileged access, 542

role of identity federation and SSO, 541–542

tools, 62

identity

DE.AE-3 framework, 221–222

- decentralized, 515–516, 518–520
- digital, 220
- federation, 541–542
- management, 221
- personal, 220
- social, 220
- verification, 219
- IDMZ (industrial DMZ), 696–697**
- IDS/IPS (intrusion detection systems/
intrusion prevention systems), 397,
539**
- IGPs, 190**
- image security, 448**
- immutability, 391**
 - blockchain, 517–518
 - developer access logs, 532–534
 - infrastructure, 461
 - microservices, 392
- incident response, 7, 20, 564**
 - during cloud migration, 646–647
 - on-premises vs. cloud, 23
 - updating, 647
- industrial networking, 691–692**
 - CPwE (Converged Plantwide
Ethernet) architecture, 695–696
 - industrial DMZ, 696–697
 - pillars of ZTNA, 696
 - creating granular trust zones
using microsegmentation,
702–705*
 - security foundation with
firewalls, 696–698*
 - visibility with network as a
sensor, 698–702*
 - Purdue Model, 693–696
 - secure remote access with ZTNA,
706–709
- infinity racetrack, 767–768**
- infrastructure. *See also* IaaS
(Infrastructure as a Service)**
 - automation, 446
 - as code, 405–406
 - immutability, 461
 - immutable, 389
 - on-premises, 576
 - resilience, 690
 - security, 396–397
- ingress controllers, Kubernetes
integration, 409–411**
- injection attacks, in cloud-native
environments, 444**
- inline tagging, 92–94**
- input validation, 457**
- insider threats, 4**
- integrity, 594**
- interface bounce template, 171**
- interference, wireless network, 345**
- Internet, 185–186, 217–218**
- IOS XE and IOS-XR**
 - commands
 - debug ip routing, 283*
 - ip cef load-sharing algorithm,
303*
 - no port-channel standalone-
disable, 162*
 - show bfd neighbors detail, 286*
 - show cts environment-data,
88–89*
 - show cts role-based counters,
313*
 - show cts role-based stg-map
all, 89–90*
 - show cts role-based stg-map
vrf, 97, 308–309*
 - show cts sxp connections brief,
311–312*

- show ip cef exact-route*, 181–182
- show ip cef vrf ENT exact-route*, 301
- show ip dhcp snooping*, 111
- show ip route super*, 282, 283
- show ipv6 interface*, 118
- show lisp instance-id*, 302
- show netconf-yang ssh server command*, 770
- show run*, 97, 759–761
- show run all*, 238
- local credentials, 224–225
- MACsec, 276–278
- quantum-safe IPsec, 273–276
- verification of SGTs, 88–91
- IoT (Internet of Things)**, 17–18, 186, 221
- IP address/ing**, 103. *See also* dynamic addressing
 - campus network, 129–130
 - management IP addresses*, 131
 - overlay user IP addresses*, 131–132
 - underlay infrastructure*, 130–131
 - maintaining continuity, 132–134
 - managing during cloud migration, 637–639
 - manual assignment, 103
- ip cef load-sharing algorithm command**, 303
- iPerf**, convergence testing, 290–296
- IPFS (InterPlanetary File System)**, 687–688
- IPsec**
 - quantum-safe, 273–276
 - TrustSec data encapsulation, 95
- IPv4**, 129–130
- IPv6**, 115–117
 - address assignment, 119–120
 - SLAAC, 119–120
 - SLAAC +DHCPv6, 120
 - stateful DHCP, 120–122
 - First Hop Security, 123
 - DHCPv6 Guard, 123–124
 - IPv6 Destination Guard, 124–125
 - ND Suppress Multicast, 126
 - Prefix Guard, 125
 - RA throttling, 125–126
 - rogue RA, 123
 - Source Guard, 125
 - GUA (Global Unicast Address), 116
 - link-local address, 116
 - neighbor discovery, 117–118
 - anycast address*, 119
 - solicited-node address*, 118
 - transient multicast addresses, 117
 - ULA (Unique Local Address), 116
 - well-known multicast addresses, 117
- ISECOM (Institute for Security and Open Methodologies)**, 3
- IS-IS (Intermediate System to Intermediate System)**, 196–198, 281
 - redistribution, 197–198
 - routing policies, 198
 - scale considerations, 198
- ISO/IEC 27001**, 14, 25
- ISP (Internet service provider)**, 130, 217–218
- IT (information technology)**, 692

J

Jenkins, 388
 Jericho Forum, 2
 JupiterOne, 416
 just-enough access, 6

K

key rotation, 401
 Keycloak SPI, 523, 524
 Kindervag, John, *No More Chewy Centers: Introducing the Zero Trust Model of Information Security*, 2
 KMS (key management systems)
 in the cloud, 400–402
 HashiCorp Vault, 402–404
 KSPM (Kubernetes Security Posture Management) tools, 449
 Kubeflow, 512
 Kubernetes, 390
 authorization flow, 519–520
 authorization module, 390
 container orchestration, 387–388
 defining and assigning a read-only role for pods, 454–455
 integrating with blockchain for smart contract-based access control, 523–530
 integration with OSS ALBs and ingress controllers, 409–411
 RBAC policies, 454–455
 retrieving and injecting secrets, 456
 webhooks, 522

L

LACP, usage with PnP, 162–163
 LAN automation, 136–138
 API-based provisioning, 143–144
 design stage, 139
 discovery stage, 139–140
 partial automated deployment, 144–147
 planning stage, 138–139
 provisioning stage, 140–143
 LAN-to-cloud microsegmentation, 100
 large-scale SDN deployment. *See also* Fast Burger SDN deployment
 automation, 369–370
 CI/CD pipeline, 373–376
 Cisco Secure Connect, 365–366
 configuration management, 371–373
 dynamic VLAN assignment, 368–369
 ISE integration, 368
 Layer 2, 364–365
 Layer 3, 365
 Meraki as Code, 370
 MX security features, 367–368
 network design, 362–363
 physical hardware, 363–364
 security, 367
 security configuration, 369
 using Git for configuration management, 371–373
 Layer 2
 design best practices, 340–342
 Fast Burger network, 364–365
 Layer 3
 design best practices, 343

- Fast Burger network, 365
- routed networks, convergence, 281–284
- least-privilege access, 6, 12, 17, 31–32, 587
- JIT access, 6
- role-based, 6
- link-local address, 116
- LISP (Locator ID Separation Protocol), 96–97, 103
- LLM (large language model), 50, 484–485
- load balancers, 215
 - application, 406–409
 - network, 406
 - open-source, 410–411
- load sharing, 214–215
- local authentication, 224–225
- local breakout, 350–351
- lock-and-key access, 6
- logs and logging
 - audit, 532–534
 - cloud-native environments, 421–422, 445
 - RADIUS, 230
 - unified, 563

M

- MAC address spoofing, 237–238
- MAC authentication bypass, 236–239
- macrosegmentation, 18, 64, 76, 77, 404. *See also* SGTs (security group tags)
 - best practices, 85–88
 - failure domain, 82
 - firewall audit trail, 81–82
 - VRF, 77–80
 - routing paths*, 80
 - stateful inspection*, 81
- MACsec, quantum-safe, 276–278
- management IP addresses, campus network, 131
- management plane
 - ACI, 209
 - BGP-EVPN, 204
 - MPLS-VPN, 201
 - SD-Access, 207
 - SD-WAN, 211
- man-in-the-middle attack, 112–113
- maturity, network, 34–35
- McEliece cryptosystem, 277
- MDM (mobile device management), 18
- measuring, network performance, 621–625
- Meraki. *See* Cisco Meraki
- Meraki as Code, 370, 371–376
- metrics, resilience, 667–671
- MFA (multifactor authentication), 5–6, 18, 31, 234–235
- microsegmentation, 18, 64, 76–77, 82–83, 404
 - Adaptive Policy, 369
 - best practices, 85–88
 - configuration policies, 83–85
 - creating granular trust zones, 702–705
 - LAN-to-cloud, 100
 - secure service insertion, 98–99
 - SGTs (security group tags), 82–83
 - GRE encapsulation*, 94
 - inline tagging*, 92–94
 - IPsec encapsulation*, 95

- methods of transport*, 91–92
- verification on IOS XE platforms*, 88–91
- VXLAN encapsulation*, 94–95
- synergy with zero trust, 490–491
- microservices, 387, 392, 409, 439
- Microsoft Azure. *See* hyperscaler cloud environments
- migration. *See also* deployment and migration phases, SDN lifecycle; LAN automation; segmentation, migration strategies; workload mobility/migration
 - 7 R's model, 583–586
 - compatibility issues, 601
 - data format and structure*, 601–602
 - legacy systems vs. modern cloud infrastructure*, 601
 - downtime risks, 573, 601
 - hybrid data, 597
 - motivations for, 574
 - access to advanced computing capabilities*, 575
 - cost efficiency and budget management*, 575
 - enhanced security and compliance*, 575–576
 - focus on core business activities*, 576
 - global reach and market expansion*, 576
 - innovation-friendly and fast development environment*, 575
 - need for, 604–605
 - risks, 605–606
 - secure data transfer
 - best practices*, 597–598
 - network*, 595–596
 - offline*, 597
 - online*, 595
 - security, 604, 605
 - tools, 603
- ML (machine language). *See* AI/ML
- modern approach to segmentation, 488–489, 572–573
- modernizing the application, 392
- module
 - NETCONF Access Control, 770–772
 - zero trust, 68
- monitoring, 20
 - authorization health, 248–249
 - Cisco ISE, 228
 - Cisco TrustSec, 354–356
 - cloud migration, 645
 - cloud-native environments, 421–422, 445
 - continuous, 22
 - convergence, 308–314
 - network, 216–217
 - regulatory requirements and compliance, 424
 - resource usage, 572
 - RF environment, 347
 - roaming behavior, 344–345
 - runtime security, 446, 456–457
 - SDN environment, 331–332
 - tools, 616–618
- MPLS (Multiprotocol Label Switching), 189
- MPLS-VPN, 200
 - control plane, 200
 - data plane, 200–201
 - management plane, 201
 - policy plane, 201

MSRB (multi-site remote border),
351–353

multicast

transient addresses, 117
well-know addresses, 117

multicloud environments, 413–414,
576–577

Cisco ASM. *See* Cisco ASM

full-stack visibility, 513

IAM (identity and access management), 542–543

policies, 414

segmentation, 491, 506–509

addressing, 493

dynamic workloads, 492–493

inconsistent policy frameworks, 492

visibility and compliance, 492

N

NAC (network access control), 8

NACM (NETCONF Access Control Module), 770–772

NAT anomaly detection, 246–248

NCSC (National Cyber Security Centre) Zero Trust Architecture Design Principles, 14

ND Suppress Multicast, 126

NDO (Nexus Dashboard Orchestrator), 508–509

Nehemiah, 219

neighbor discovery, IPv6, 117–118

anycast address, 119

solicited-node address, 118

NETCONF, 43–44, 90, 75, 756,
761, 770

Access Control Module, 770–772

service-level restrictions, 769–770

NetDevOps, 40–41, 172

CI/CD/CT (Continuous Integration/
Continuous Development/
Continuous Testing), 41

IaC (Infrastructure as Code), 41–46

**network/s, 185. *See also* industrial
networking; overlay; SDN
(software-defined networking);
segmentation; underlay**

access control, 235

automation. *See* automation, network
best practice design concepts, 338–
339

campus. *See* campus network

“as code” deployment, 772–775

convergence, 279–280. *See also*
convergence

in data center networks,
290–296

in Layer 3 routed architectures,
281–284

routing protocol dependency,
280–281

server-side verification,
288–290

in software-defined architectures,
296–300

testing, 300–303

timers, 284–288

data transfers, 595–596

design, 213

fully compliant, 137

load balancer, 406

management protocols, 43–44

manual configuration, 746

maturity, 34–35

monitoring, 216–217

neural, 49

- outages, 187
 - performance
 - measuring*, 621–625
 - monitoring and optimization*, 636
 - policies, 469, 543–544
 - resilient, 193–194
 - security, 396–397, 616
 - segmentation, 7–8, 20, 64–65, 75–76, 618–619
 - macro-*, 64, 77–80, 404
 - micro-*, 64, 82–86, 404
 - VRF*, 77–80
 - tiered, 340
 - cabling*, 340
 - financial sector use case*, 358–359
 - Layer 2*, 340–342
 - Layer 3*, 343
 - port channels*, 340
 - stacking constructs*, 342
 - upgrades, 128, 188–189
 - uptime, 187, 189
 - virtual, 77, 189
 - virtual private, 66
 - visibility, 9
 - VLANs (virtual LANs), 75
 - wireless
 - containment*, 346
 - interference*, 345
 - monitoring client roaming behavior*, 344–345
 - RF environment*, 347
 - salt-and-pepper design*, 344
 - neural network, 49
 - “never trust, always verify” approach, 4–5
 - NFV (network function virtualization)
 - orchestration, 30
 - NIST (US National Institute of Standards and Technology), 25
 - DE.AE-3 framework, 221–222
 - SP 800–207, 11, 12, 432
 - NLB (network load balancer), features and use cases, 408–409
 - NMS (network management system), 33
 - no port-channel standalone-disable command, 162
 - nodes, 44–45
 - northbound API, 35
- ## O
-
- objects, serialization, 42–43
 - observability, 390, 422, 448. *See also* monitoring; OpenTelemetry; visibility
 - cloud-native, 627
 - integrating into cloud migration, 628–629
 - OCSP (Online Certificate Status Protocol), 232
 - offline data migration, 597
 - online data migration, 595
 - OPA (Open Policy Agent), 512
 - Open Tofu, 767
 - open-source, tools, 511, 618. *See also* OSS (open-source software)
 - OpenTelemetry, 422–423
 - integrating into network measurements, 632–634
 - role in cloud migration, 630–632
 - operations and management phases, SDN lifecycle, 330

- integrating SDN automation with existing IT operations management systems, 330–331
 - maintaining security and compliance, 334–335
 - monitoring and troubleshooting the SDN environment, 331
 - developing troubleshooting procedures, 333–334*
 - implementing comprehensive monitoring tools, 331–332*
 - utilizing SDN controller’s visibility features, 332–333*
 - OpEx, 572, 775
 - Option 15, 157
 - Option 43, 157, 159–160
 - Option 60, 157–159
 - Option 82, 107–109
 - Option 90, 115
 - options, DHCP, 113
 - orchestration, 9, 30–31
 - Ansible
 - nodes, 44–45*
 - playbooks, 44–45*
 - APIs, 45
 - cloud-native environments, 447
 - container, 387–388
 - cross-domain, 510–513
 - NFV (network function virtualization), 30
 - service, 30
 - Terraform, 46
 - workflow, 30
 - organizations
 - maturity, 34
 - processes, 17
 - OSI model, 186
 - OSPF (Open Shortest Path First), 192–193
 - areas, 193
 - cost, 193
 - redistribution, 194
 - routing policies, 195
 - scale considerations, 195–196
 - OSS ALBs, Kubernetes integration, 409–411
 - OT (operational technology), 692
 - outages, 187
 - overlay, 128, 189
 - protocols. *See also* ACI (Cisco Application Centric Infrastructure); SD-Access; SD-WAN
 - BGP-EVPN, 201–204*
 - MPLS-VPN, 200–201*
 - SD-Access, 299–300
 - user IP addresses, 131–132
 - OWASP (Open Worldwide Application Security Project), 36, 406, 440
 - comparing Top 10 Web Application vs. Cloud-Native Application Security Risks, 440–443
 - SAMM (Software Assurance Maturity Model). *See* SAMM (Software Assurance Maturity Model)
 - ServerlessGOAT, 475
-
- P**
- PaaS (Platform as a Service), 580
 - PaC (Policy as Code), 424
 - PAM (privileged access management), 651–652

- PAP (Password Authentication Protocol), 223–224
- partial automated deployment, 144–147
- password/s, 5–6, 220–221, 457–458.
See also authentication
- patching, 250–252
- on-path attack, 236–237, 239
- PCI DSS (Payment Card Industry Data Security Standard), 26
- PEN (Policy Extended Node), SD-Access, 712–715
- penetration testing, 397
- PEPs (policy enforcement points), 8
- performance
 - application, 626
 - network
 - measuring*, 621–625
 - monitoring and optimization*, 636
- perimeter-focused security model, 2
- permissions, 654–656
- physical resilience, 671–672
- planning, campus network, 128–129
- planning and design phases, SDN lifecycle, 318
 - defining network requirements and use cases, 318–319
 - designing a robust network topology, 322–323
 - planning for multidomain and cloud integration, 324
 - reviewing key characteristics of SDN controllers, 321
 - security considerations, 323–324
 - selecting the right SDN architecture, 319
 - on-premises controller model*, 319–320
 - SaaS controller model*, 320–321
- planning stage, LAN automation, 138–139
- platform/s
 - automated analysis, 663–664
 - engineering, 679–680
 - security, 448–450
 - SOAR, 540–541
- playbooks, 44–45
- Plug-and-Play Connect, 161–162. *See also PnP (Plug-and-Play)*
- PnP (Plug-and-Play), 149, 150. *See also ZTP (zero-touch provisioning)*
 - agent, 151
 - authorization of devices, 163–164
 - certificates, 151–154
 - Cisco Catalyst Center call flow, 151–152
 - FQDN, 151, 153
 - global deployment, 180–183
 - LACP usage, 162–163
 - Meraki onboarding flow, 164–165
 - pan-Africa deployment, 177–180
 - script, 179–180
 - server, 151–152
 - time management, 155–156
 - using DHCP, 156–157
 - DHCP server scope*, 157
 - Option 6*, 157
 - Option 15*, 157
 - Option 43*, 159–160
 - Option 60*, 157–159
 - using DNS, 160–161
- policy plane
 - ACI, 209
 - BGP-EVPN, 203–204

- MPLS-VPN, 201
- SD-Access, 206
- SD-WAN, 211
- policy/ies, 17, 20. *See also* Common Policy; PEPs (policy enforcement points)
 - application access, 59–60
 - based remediation, 564
 - BYOD, 56–57, 58
 - as code, 424
 - configuration, 83–84
 - data protection and privacy, 543
 - drift, 494
 - dynamic, 245
 - end-to-end, 718. *See also* end-to-end segmentation
 - end-to-end segmentation, 493
 - enforcement, 23, 54, 459, 494, 530–532
 - CASBs (Cloud Access Security Brokers), 539–540*
 - firewalls and intrusion detection/intrusion prevention systems, 539*
 - SOAR, 540–541*
 - firewall, 735–739
 - frameworks, 492, 537
 - IAM, 453–454, 541
 - managing privileged access, 542*
 - role of identity federation and SSO, 541–542*
 - incident response, 564
 - models
 - Cisco ACI, 498*
 - cloud, 499–500*
 - comparison and mapping of constructs, 500–501*
 - comparison to cloud segmentation models, 501*
 - multicloud security, 414
 - network, 469
 - network security, 543–544
 - predictive and adaptive, 565
 - principles of effective management, 536
 - RADIUS, 244–245
 - rate limiting, 467
 - RBAC, 454–455
 - regulatory compliance, 543
 - routing
 - EIGRP, 192*
 - OSPF, 195*
 - security, 535–536
 - designing, 536–538*
 - governance, 543*
 - segmentation, 76
 - automation, 510–513*
 - consistency across hybrid and multicloud environments, 506–509*
 - migrating, 496*
 - server, 245
 - traffic engineering, 217
 - unified, 495–496
- port channels, 340
- post-migration security maintenance, 613–614
- post-quantum cryptography, 265–266, 270
- PPDIOO model, 317–318. *See also* SDN lifecycle
- Prefix Guard, 125
- on-premises. *See also* cloud environments; hybrid IT infrastructure

controller model, 319–320
 infrastructure, 576
 zero trust
 continuous monitoring and threat detection, 22
 encryption and data protection, 22
 explicit verification, 20–21
 incident response, 23
 least-privilege access, 21
 policy enforcement, 23
 segmentation, 21
 SOAR (*security orchestration, automation, and response*), 22

pre-validation, 775–776
 principle of least privilege, 460–461.
 See also least-privilege access

privacy
 BYOD policy and, 58
 policy, 543

Private 5G, 348–350

private cloud, 397

process mining, 757

programmability-based deployment, 172

Prometheus, 512

protocol/s
 BB84, 271–272
 BFD (bidirectional forwarding detection), 285–286
 classless, 194
 connectionless, 196
 E91, 271
 network management, 43–44
 routing. *See* routing protocols
 security, 298, 769
 synchronization, 281

provider-consumer model, 502

provisioning, 446
 “as code”, 772–775
 infrastructure automation, 446
 NETCONF service-level restrictions, 769–772
 security and compliance enforcement, 446

provisioning stage, LAN automation, 140–143

public cloud, 397

Pulumi, 510, 767

Purdue Model, 693–696

Python, script, 751–752

Q

QE (quality engineering), 614
 ensuring data security and navigating cloud migration with precision, 615
 landscape of cloud migration challenges, 614
 strategic blueprint, 614–615

QKD (quantum key distribution), 270, 271–272

QoS (quality of service), 215

quantum computing, 4, 253, 264–265. *See also* security, safeguarding against quantum adversaries
 Bra-ket notation, 256–257
 circuits, 264
 CV-QKD (Continuous Variable QKD), 271
 decoherence time, 258
 DiVincenzo criteria, 258
 emerging security threats, 265–266, 269

- entanglement, 258–259
- Grover’s algorithm, 269
- QKD (quantum key distribution), 271–272
- quantum gates, 260–261
 - CNOT gate*, 262–264
 - H gate*, 261–262
- qubits, 255–256, 257–258
- Shor’s algorithm, 268–269
- superposition, 256–257, 261–262
- quantum-safe IPsec**, 273–276
- qubits**
 - Bell state, 263–264
 - Bra-ket notation, 256–257
 - entanglement, 258–259
 - modalities, 257–258

R

- RADIUS (Remote Authentication Dial-In User Service)**, 225–227
 - authorization, 243–245
 - logs, 230
 - monitoring, 248–249
 - policies, 244–245
- RAG (retrieval-augmented generation)**, 50
- RARP (Reverse Address Resolution Protocol)**, 104
- rate limiting, API**, 37, 467
- RBAC (role-based access control)**, 468–469
- reconnaissance attack**, 39
- redistribution**
 - EIGRP, 191–192
 - IS-IS, 197–198
- regions, cloud**, 572
- regulatory compliance**, 424, 460, 543, 575–576, 607
- reinforcement learning**, 48
- reliability of auditable data**, 688–689
- resilience/resiliency**, 193–194
 - audit trail, 686–687
 - authentication and authorization, 681–684
 - client and server, 684–686
 - in the cloud, 676–681
 - cloud-native, 389–390
 - domain, 672–673
 - environmental redundancy, 672
 - metrics, 667–671
 - network infrastructure, 690
 - physical, 671–672
 - software, 674
 - validation, 689–690
 - vendor, 673–674
- resource/s**
 - limiting, APIs, 469–470
 - structuring in the cloud, 653–654
 - usage, monitoring, 572
- REST-API authentication methods**, 233
- RESTCONF**, 43–44, 749–750, 753–756, 759–761
- revision control**, 761
 - event manager-triggered configuration backups, 762–763
 - Git configuration for archiving changes, 763
 - using configuration archive, 761–762
- RF environment**
 - choice of medium, 348–350
 - monitoring, 347
 - spread spectrogram, 347

utilization, 347

RFC 3118, DHCP authentication, 114–115

risk score, 700–701

rogue DHCP servers, 109–111

rogue RA, 123

ROI (return on investment), 581

role-based access, 6

rollback timer, 752–753

routing

- application-aware, 70–71
- geo-, 218
- policies
 - EIGRP*, 192
 - IS-IS*, 198
 - OSPF*, 195
- SDN (software-defined networking), 188

routing protocols, 186, 187–190, 214, 281–282

- BGP, 190
- convergence, 280–281, 284–288
- EIGRP, 190–192
 - redistribution*, 191–192
 - routing policies*, 192
 - scale considerations*, 192
- IGPs, 190
- IS-IS, 196–198, 281
 - redistribution*, 197–198
 - routing policies*, 198
 - scale considerations*, 198
- MPLS-VPN, 200
- OSPF, 192–196
 - areas*, 193
 - cost*, 193
 - redistribution*, 194
 - scale considerations*, 195–196

- overlay, 198–200
- synchronization, 281

RSA (Rivest-Shamir-Adleman) algorithm, 267

RTT (round-trip time), 227

runtime, 446

- confidential computing, 447
- security monitoring, 446, 456–457

S

SaaS (Software as a Service), 320–321, 580

salt-and-pepper design, 344

SAMM (Software Assurance Maturity Model), 557–558

- activities, streams, and maturity levels, 559
- benefits and implementation, 562–563
- business functions, 558–559
- relationship to software development life cycles, 561–562

SAST (static application security testing), 462

SBOM (software bill of materials), 465–466

scale considerations

- EIGRP, 192
- IS-IS, 198
- OSPF, 195–196

score, trust, 58

script. *See also automation*

- based automation, 33
- PnP, 179–180
- Python, 751–752
- TCL, 748–749
- toil, 756–757

- SD-Access, 204–205, 703**
 - control plane, 205–206
 - data plane, 206
 - EN (Extended Node), 711–712
 - extending ZTNA to noncarpeted environments, 710–711
 - management plane, 207
 - MSRB (multi-site remote border), 351–353
 - in overlay architectures, 299–300
 - PEN (Policy Extended Node), 712–715
 - policy plane, 206
 - SDK (software development kit), 45**
 - SDL (Secure Development Lifecycle), 547–549, 555–557**
 - SDLC (Software Development Life Cycle), 544–545, 546, 553–557**
 - SDN (software-defined networking), 33–34, 315. *See also* third-party SDN integrations**
 - adoption challenges, 316–317
 - convergence, 296–300
 - routing, 188
 - troubleshooting, 333–334
 - underlay routing protocols, 188–190
 - SDN lifecycle**
 - deployment and migration phases, 324
 - deploying the SDN controller, 326–329*
 - preparing the network infrastructure, 325–326*
 - operations and management phases, 330
 - integrating SDN automation with existing IT operations management systems, 330–331*
 - maintaining security and compliance, 334–336*
 - monitoring and troubleshooting the SDN environment, 331–334*
 - planning and design phases, 318
 - defining network requirements and use cases, 318–319*
 - designing a robust network topology, 322–323*
 - planning for multidomain and cloud integration, 324*
 - reviewing key characteristics of SDN controllers, 321*
 - security considerations, 323–324*
 - selecting the right SDN architecture, 319–321*
- SDNC (SDN Controller), 315. *See also* Cisco Catalyst Center**
 - key characteristics, 321
 - on-premises, 319–320
 - SaaS, 320–321
 - visibility features, 332–333
- SD-WAN, 209–210**
 - based SSE integration, 70–71
 - control plane, 210
 - data plane, 210–211
 - DIA (Direct Internet Access), 214
 - management plane, 211
 - policy plane, 211
- secrets management, 461, 464**
- secure data transfer, 597–598**
 - best practices, 597–598
 - versus cloud migration, 598–599
 - Direct Connect Services, 596
 - hybrid, 597

- impact of bandwidth limitations, 599–600
- network, 595–596
- offline, 597
- online, 595
- security, 217. *See also* authentication; CSPM (cloud security posture management); encryption; threat/s
 - ACLs, 404–405
 - API, 36–38, 459, 467–470
 - application. *See* cloud-native, application security
 - assessments, 397
 - automation, 405–406
 - breach/es, 4–6
 - assumption of*, 7
 - containment*, 7
 - in the cloud, 382
 - cloud, 383, 400–402. *See also* cloud environments, security
 - cloud migration, 604, 605
 - cloud-native, 390–391, 411, 445. *See also* 4C’s of Cloud-Native Security; CNAPP (cloud-native application protection platform)
 - app definition and development*, 447–448
 - authentication and access control in distributed systems*, 444
 - Cisco Secure technologies*, 419–421
 - CNCF projects*, 450–453
 - core principles*, 391–392
 - HashiCorp Vault*, 402–404
 - injection and misconfiguration risks*, 444
 - logging and monitoring*, 445
 - managing component vulnerabilities as scale*, 444–445
 - microservices*, 392
 - observability and analysis*, 448
 - Open Telemetry*, 422–423
 - orchestration and management*, 447
 - platforms*, 448–450
 - provisioning*, 446
 - runtime*, 446–447
 - Splunk*, 423
 - CTEM (continuous threat exposure management), 427–430
 - data, 593
 - de-perimeterization, 2
 - by design, 396
 - DHCP, 107
 - endpoint, 8–9, 20, 23, 65–66
 - enforcement points, Cisco TrustSec, 356
 - image, 448
 - immutability, 391
 - infrastructure, 396–397
 - multicloud, 413–414
 - network, 396–397
 - patches, 250–252
 - perimeter-focused model, 2
 - policy/ies, 535–536
 - designing*, 536–538
 - frameworks*, 537
 - governance*, 543
 - posture, 9, 65–66
 - protocols, 298
 - rulesets, 169
 - safeguarding against quantum adversaries, 270

- practical solution approach*, 272–273
- quantum-safe IPsec*, 273–276
- quantum-safe MACsec*, 276–278
- SDN controller, 323–324
- SDN environment, 334–335, 367
- serverless, 471–475
- threat landscape, 3–5
- tools, 18
- trust, 2, 3
- Web3, 517
- segmentation**, 7–8, 20, 64–65, 76, 396, 487–488, 618–619.
See also Cisco TrustSec;
macrosegmentation;
microsegmentation
- API, 60
- application-level, 60
- challenges in hybrid and multicloud environments, 491
 - addressing*, 493
 - dynamic workloads*, 492–493
 - inconsistent policy frameworks*, 492
 - visibility and compliance*, 492
- cloud-native architectures, 496–497
- data center, 571
- end-to-end, 493, 718, 719–723
- environment, 60
- governance considerations, 77
- macro-, 64, 76–77, 404
 - best practices*, 85–88
 - VRF*, 77–80
- micro-, 18, 64, 76–77, 82–86, 404, 490–491
 - best practices*, 85–88
 - configuration policies*, 83–85
 - creating granular trust zones*, 702–705
- LAN-to-cloud*, 100
- secure service insertion*, 98–99
- SGTs (*security group tags*), 82–83
- migration strategies, 505
 - automated policies for rapid “big bang” migrations*, 506
 - parallel deployment for hybrid continuity*, 505
 - phased migration with policy layers*, 505
- modern approach, 488–489, 572–573
- policies, 76
 - automation*, 510–513
 - consistency across hybrid and multicloud environments*, 506–509
 - migrating*, 496
- software-based, 76
- traditional, 488, 571–572
- workload-specific, 494
- serialization**, 42–43
- server**
 - DHCP, 157
 - PnP, 151–152
 - policies, 245
- serverless architectures**, 470, 580
 - attack vectors, 475–476
 - key security challenges, 471–472
 - security best practices, 472–475
 - security flow, 477–482
 - security-first approach, 475
 - shared responsibility model, 471
- service/s**
 - accounts, 228–231

- Encryption as a, 403
- Firewall as a, 67
- Function as a, 581
- Infrastructure as a, 579–580
- micro-, 387, 392, 439
- network and access control, 398–400
- orchestration, 30
- to-service communication, 392–393
- Software as a, 580
- SGACLs, 309–310**
- SGTs (security group tags), 77, 82–83, 132. *See also* Cisco TrustSec; end-to-end segmentation**
 - for end-to-end segmentation, 718
 - GRE encapsulation, 94
 - for industrial network segmentation, 703–705
 - inline tagging, 92–94
 - IPsec encapsulation, 95
 - methods of transport, 91–92
 - priority order, 98
 - verification on IOS XE platforms, 88–91
 - VXLAN encapsulation, 94–95
- shared responsibility model, 393–394, 471**
- sheep-dip, 706**
- Shor’s algorithm, 268–269**
- show bfd neighbors detail command, 286**
- show cts environment-data command, 88–89**
- show cts role-based counters command, 313**
- show cts role-based stg-map all command, 89–90**
- show cts role-based stg-map vrf command, 97, 308–309**
- show cts sxp connections brief command, 311–312**
- show ip cef exact-route command, 181–182**
- show ip cef vrf ENT exact-route command, 301**
- show ip dhcp snooping command, 111**
- show ip route super command, 282, 283**
- show ipv6 interface command, 118**
- show lisp instance-id command, 302**
- show netconf-yang ssh server command, 770**
- show run all command, 238**
- show run command, 97, 759–761**
- SIEM (security information and event management), 18, 35, 38–40**
- site type template, 134**
- SLA (service-level agreement), 187, 753–754**
- SLAAC (Stateless Address Autoconfiguration), 119–120**
- smart contracts, 517, 523–530**
 - coding mistakes and exploits, 520–521
 - for policy enforcement, 530–532
 - revocation challenges, 522
- smart devices, 32**
- SMS (Short Message Service), 234**
- Snyk, 388**
- SOAR (security orchestration, automation, and response), 18–19, 22, 35, 540–541**
 - dynamic playbook, 39–40
 - integration with SIEM and XDR, 38–40
- SOC 2 (System Organization Controls 2), 25**

- social engineering, 16
- social identity, 220
- software. *See also* SDN (software-defined networking)
 - based segmentation, 76
 - bill of materials, 445, 465–466
 - in-house vs. COTS (commercial off-the-shelf), 675–676
 - open-source, ALBs, 409–411
 - patching, 250–252
 - resilience, 674
 - revision control, 761
 - event manager-triggered configuration backups*, 762–763
 - Git configuration for archiving changes*, 763
 - using configuration archive*, 761–762
 - supply chain, 465–467
 - versioning, 165, 674–675
 - vulnerabilities, causes of, 545–546
 - zero trust, 19
- software development, framework/s, 553–557. *See also* DevOps; DevSecOps
- solicited-node address, 118
- Source Guard, 125
- southbound API, 35
- Splunk, 423, 426
- spread spectrogram, RF utilization, 347
- Spring Boot, 387
- SRA (Cisco security reference architecture), 15
- SSDLC (Secure Software Development Lifecycle), 550–552, 554–557
 - benefits, 552–553
 - phases, 552
- SSE (Security Service Edge), 67–68
 - deployment approaches
 - client-based*, 68–69
 - clientless*, 69–70
 - SD-WAN-based integration, 70–71
 - VPN-based integration, 70
- SSID (service set identifier)
 - monitoring, 345–346
 - spoofed, 345
- SSO (single sign-on), 59, 541–542
- standard template, 169–170
- standards, 14
 - cloud-native, 430–431
 - FIPS (Federal Information Processing Standards), 24–25
 - IEC 62443, 698
 - ISO/IEC 27001, 14, 25
 - NIST (US National Institute of Standards and Technology), 25
 - PCI DSS (Payment Card Industry Data Security Standard), 26
 - zero trust
 - CARTA (Continuous Adaptive Risk and Trust Assessment) model*, 12
 - CISA (Cybersecurity and Infrastructure Security Agency) Zero Trust Maturity Model*, 13
 - Forrester’s Zero Trust eXtended (ZTX) framework*, 12
 - NCSC (National Cyber Security Centre) Zero Trust Architecture Design Principles*, 14
 - NIST SP 800–207*, 12
 - shared features and functions*, 14–15

stateful inspection, 81
 stateful packet generation, 304–305
 stateless traffic generation, 305–307
 storage

- blockchain-based, 687–688
- data lake, 50–51
- password, 457–458
- secrets, 461

 STP (Spanning Tree Protocol), 131–132, 340
 StratoShark, 473
 structured data, 759–761
 subjects, 502–503
 superposition, 256–257, 261–262
 supervised learning, 48
 supply chain

- component integrity, 466–467
- dependency tracking, 466
- securing trusted components, 466
- security, 448
- software, 465–467
- vendor platforms, 450

 SWG (Secure Web Gateway), 67
 SXP (Scalable Group Tag Exchange Protocol), 96
 symmetric encryption, 266

- CV-QKD (Continuous Variable QKD), 271
- quantum key distribution, 271–272

T

security group, 82–83
 TALOS, 426
 TCL script, 748–749
 TCO (total cost of ownership), 581
 template/s, 169–170

- composite, 170
- interface bounce, 171
- site type, 134
- standard, 169–170

 Terraform, 46, 389, 510, 766–767
 tests/testing

- automation, 29
- continuous, 41, 47
- convergence, 300–303
 - iPerf*, 290–296
 - stateful traffic generation*, 304–305
 - stateless traffic generation*, 305–307
 - traffic generation*, 303–304
- dynamic application security, 37–38, 462
- penetration, 397
- SDN, 329
- static application security, 462

 Thanos, 512
 third-party SDN integrations, 717

- challenges in multivendor environments, 719–723
- end-to-end policy strategy in a multivendor environment, 718
- firewall, 735–743
 - connectivity in the campus*, 728–734
 - highly resilient integrations*, 740–743

TACACS+, authorization, 243
 tags. *See also* SGTs (security group tags)
 OSPF, 195

- policy integration*, 735–739
 - security considerations in the campus, 727
 - VXLAN-EVPN, 723–727
- threat/s**
 - actors, 4
 - advanced persistent, 4
 - anticipating, 426–427
 - awareness training, 16–17
 - continuous detection and mitigation, 10
 - correlation and automation for containment, 705–706
 - detection, 22
 - detection and response, 398–400
 - insider, 4
 - landscape, 3–5
- tiered network design**, 340
 - cabling, 340
 - financial sector use case, 358–359
 - Layer 2, 340–342
 - Layer 3, 343
 - port channels, 340
 - stacking constructs, 342
- time-based network authorization**, 250
- toil**, 756–757
- tools**. *See also* Ansible; Terraform
 - automated remediation, 464–465
 - cloud management, 647–649
 - configuration management, 464
 - Crossplane, 510
 - Expect, 746–747
 - Falco, 456–457
 - IaC, 766–769
 - Jenkins, 388
 - KSPM (Kubernetes Security Posture Management), 449
 - migration, 589–590, 592–593, 603
 - AWS Migration Hub*, 590
 - Azure Migrate*, 590
 - Google Cloud Migration Center*, 590
 - monitoring and assessment, 616–618
 - multicloud security, 413–414
 - network performance measurement, 624–625
 - observability, 626–627
 - open-source, 409–411, 511
 - OpenTelemetry
 - integrating into network measurements*, 632–634
 - role in cloud migration*, 630–632
 - SDN monitoring, 331–332
 - secure coding, 462
 - security, 18
 - SOAR (security orchestration, automation, and response), 18–19, 22
 - Terraform, 389
 - trust tolerance management, 62–63
 - UEBA (user and entity behavior analytics), 580
 - user access management, 656
 - Vault, 390
- Top 10 Security Risks for Cloud-Native Environments**, 440–443
- traceability**, 587
- traditional segmentation**, 488, 571–572
- traffic captures**, 289
- traffic engineering**, 212–213
 - bandwidth planning, 216
 - geo-routing, 218
 - global Internet, 217–218
 - load balancing and sharing, 214–215

- network design, 213
- network monitoring and optimization, 216–217
- policy and security, 217
- QoS (quality of service), 215
- routing protocol selection, 214
- traffic flow analysis, 214
- traffic management, 214
- traffic generation, 303–304**
 - stateful, 304–305
 - stateless, 305–307
- training**
 - AI/ML, 48, 50
 - threat awareness, 16–17
- transient multicast addresses, 117**
- tromboning, 80**
- troubleshooting, SDN environment, 333–334**
- trust, 2, 3**
 - base, 60–61
 - certificate-based, 57
 - decentralized, 518. *See also* Web3
 - device, 56–58
 - levels, 55
 - score, 58
 - tolerance, 62–63
 - user, 55–56. *See also* user trust
 - collecting user data, 61–62*
 - verifying, 56*
- TrustSec. *See* Cisco TrustSec**
- IP addressing, 130–131**
 - routing protocols, 188–190. *See also*
 - routing protocols
 - EIGRP, 190–192*
 - IS-IS, 196–198*
 - OSPF, 192–196*
- unified logging, 563**
- unified security models, 395**
- unstructured data, 758–759**
- unsupervised learning, 48**
- updating**
 - in an “as code” network deployment, 773
 - incident response, 647
- upgrades, network, 188–189**
- uptime, network, 187, 189**
- US Executive Order on Improving the Nation’s Cybersecurity, 11**
- use cases, LLM (large language model), 50**
- user**
 - identity, 5, 63–64. *See also* IAM (identity and access management)
 - permissions, 654–656
 - role, 6, 63
 - trust, 55–56
 - collecting user data, 61–62*
 - greenfield/brownfield environments, 56*
 - verifying, 56*
- UTP (unshielded twisted pair), 340**

U

- UEBA (user and entity behavior analytics) tools, 580**
- ULA (Unique Local Address), 116**
- underlay, 128**

V

- van Eck phreaking, 340**
- Vault, 390**
- vendor platforms, 450**
- vendor resilience, 673–674**

verifying

- identity, 219
- user trust, 56

versioning, software, 165**visibility**

- cloud-native environment, 422
- hybrid and multicloud environments, 513
- industrial network, 698–702
- network, 9
- SDNC (SDN Controller), 332–333
- segmentation and, 492
- serverless architectures, 473

VLANs (virtual LANs), 75**VM (virtual machine)**

- NAT anomaly detection, 246–248
- network access, 246

VN (virtual network), 77, 189, 639, 640–641**VPN (virtual private network), 66, 397****VRF (virtual routing and forwarding), 77**

- macrosegmentation, 77–80
- secure service insertion, 98–99
- stateful inspection, 81

VTP (VLAN Trunking Protocol), 137**vulnerability/ies**

- management, 459
- software, 545–546

VXLAN-EVPN, 722, 723–727**W**

web certificate, 152–154**Web3, 514–515, 518. *See also* blockchain; DID (decentralized identity)****blockchain, 517****DID (decentralized identity), 515–516****hybrid cloud challenges, 516****role in policy enforcement, 516****security, 517****smart contracts, 517****wallet, 523–524****webhooks, 522, 527****Well-Architected Frameworks, 587, 588–589. *See also* workload mobility/migration, Well-Architected Framework****well-known multicast addresses, 117****westbound API, 36****wireless networks/Wi-Fi**

- anchoring, 351–354
- containment, 346
- DHCP rogue servers, 111
- interference, 345
- local breakout, 350–351
- monitoring client roaming behavior, 344–345
- RF environment, 347–349
- salt-and-pepper design, 344
- tunneling, 350

Wireshark, filters, 289–290**WLC (wireless LAN controller), Option 82, 108–109****workflow**

- automation, 358–359
- business, 66–67
- orchestration, 30

workload mobility/migration, 571, 581. *See also* QE (quality engineering)**7 R's model, 583–586****access controls, 619–620**

- benefits and challenges, 572–574
- building out a secure migration plan, 583
- compatibility issues, 601
 - applications*, 603–604
 - data format and structure*, 601–602
 - legacy systems vs. modern cloud infrastructure*, 601
- context, 581–582
- continuous monitoring and assessment, 616–618
- data security during, 593
- deployment model, 582
- designing your cloud architecture and migration strategy, 644–645
- Direct Connect Services, 596
- DNS, 642–643
- downtime, 601
- DR (disaster recovery), 645
- ensuring data integrity and confidentiality, 593–594
- ensuring data security, 591
- frameworks and tools, 589–590, 592–593
 - AWS Migration Hub*, 590
 - Azure Migrate*, 590
 - Google Cloud Migration Center*, 590
- HA (high availability), 643
- hyperscalers, 647–649
- identifying “what” to migrate, 591
- incident response, 646–647
- integrating observability, 628–629
- maintenance of network security postures, 616
- managing IP addressing and DNS changes, 637–639
- measuring network performance, 621–625
- monitoring, 645
- moving applications and data, 582
- need for, 604–605
- network and connectivity considerations, 616, 621
- network segmentation, 618–619
- observability tools, 626–627
- online, 595
- OpenTelemetry, 630–632
- people, processes and technology, 582–583
- pre-migration security preparations, 610–612
- preparing for cloud migration security concerns, 606–607
- risks, 605–606
- ROI (return on investment), 581
- role of automation and integration, 590
- safeguards, 607
 - APIs and access control*, 607–608
 - confirming security measures*, 610
 - employing a phased migration strategy and risk mitigation*, 608
 - encrypting your data during transit*, 608
 - formulating a security plan*, 609
 - implementing decommissioning and sanitization activities*, 609
 - limiting data access during migration*, 608

maintaining data protection and integrity, 609

understanding data and compliance requirements, 607

secure data transfer, 594–595

hybrid data, 597

network, 595–596

offline, 597

security, 604

during migration, 610–612

post-migration, 613–614

posture adjustment, 645

during transition phases, 586

TCO (total cost of ownership), 581

tools, 603

Well-Architected Framework, 587

applying security at all layers, 587

building a strong identity foundations, 587

embracing the AWS Shared Responsibility Model, 588

ensuring data is protected in transit and at rest, 588

keeping people away from data, 588

maintaining traceability, 587

making security best practices into habits, 587

performing risk assessment and mitigation, 588

preparing for security incidents, 588

workload-specific segmentation, 494

X

x.509 certificate-based authentication, 231–233

XDR (extended detection and response), 35, 38–40

XSS (cross-site scripting) attack, 37

Y

YAML configuration files

LAN automation API parameter, 143

Meraki as Code, 371–373

YANG, 44

Z

zero trust, 1, 381, 489–490, 575–576. *See also* trust

across closed-loop automation, 50–51

application access policies, 59–60

benefits, 490

central inventory, 63

in the cloud. *See also* cloud; cloud-native environments

recommendations, 19–20

SSE (Security Service Edge), 67–68

for cloud native, 431–433

comprehensive strategy, 15–16

people, 16–17

processes, 17–18

technology, 18–19

core principles

assume breach, 7, 32

explicit verification, 5–6, 19, 31

least-privilege access, 6, 19, 21, 31–32

development milestones

BeyondCorp initiative, 2

ISECOM (Institute for Security and Open Methodologies), 3

- device trust, 56–58
- dynamic addressing, 109. *See also*
 - dynamic addressing
- endpoint security, 65–66
- functional pillars, 53–54
- hybrid environment, 23–24
- identifying business workflows,
 - 66–67
- industry standards, 11
 - CARTA (Continuous Adaptive Risk and Trust Assessment) model*, 12
 - CISA (Cybersecurity and Infrastructure Security Agency) Zero Trust Maturity Model*, 13
 - Cloud Security Alliance (CSA) Zero Trust Working Group*, 12–13
 - Forrester’s Zero Trust eXtended (ZTX) framework*, 12
 - ISO/IEC 27001*, 14
 - NCSC (National Cyber Security Centre) Zero Trust Architecture Design Principles*, 14
 - NIST SP 800–207*, 11, 12
 - shared features and functions*, 14–15
- as national policy, 11
- need for, 3–5
- “never trust, always verify” approach, 4–5
- relationship to CSPM and CWPP, 419
- segmentation, 60, 64–65. *See also*
 - segmentation
 - end-to-end*, 493
 - macro-*, 64
 - micro-*, 64, 490–491
 - workload-specific*, 494
- software, 19
- trust tolerance, 62–63
- unified policies, 495–496
- user trust, 55–56
 - greenfield/brownfield environments*, 56
 - verifying*, 56
- zones**, 698
- ZTA (zero trust architecture)**, 1, 30. *See also* zero trust
- ZTNA (zero trust network access)**, 397
 - brownfield deployment, 73–74
 - extending to noncarpeted environments, 710–711
 - greenfield deployment, 71–73
 - for industrial plant networks, 696
 - creating granular trust zones using microsegmentation*, 702–705
 - security foundation with firewalls*, 696
 - visibility with network as a sensor*, 698–702
 - secure remote access, 706–709
- ZTP (zero-touch provisioning)**, 136–137, 165, 166, 170
 - claiming devices
 - in Catalyst Center*, 166–168
 - in the Meraki dashboard*, 168–169
 - foundation configurations, 165–166
 - software and hardware deployment selection in Catalyst Center, 166–167