Introduction to Software-Defined Perimeter

Zero Trust Training - Training course study guide

cloud security alliance®

ZTT™ Zero Trust Training

The official location for Software-Defined Perimeter Working Group is
https://cloudsecurityalliance.org/research/working-groups/software-defined-perimeter/

Disclaimer

Cloud Security Alliance designed and created this Zero Trust Training course study guide (the "Work") primarily as an educational resource for security and governance professionals. Cloud Security Alliance makes no claim that use of any of the Work will assure a successful outcome. The Work should not be considered inclusive of all proper information, procedures and tests or exclusive of other information, procedures and tests that are reasonably directed to obtaining the same results. In determining the propriety of any specific information, procedure or test, professionals should apply their own professional judgment to the specific circumstances presented by the particular systems or information technology environment.

About Cloud Security Alliance

The Cloud Security Alliance℠ (CSA) (www.cloudsecurityalliance.org) is the world's leading organization dedicated to defining and raising awareness of best practices to help ensure a secure cloud computing environment. Cloud Security Alliance harnesses the subject matter expertise of industry practitioners, associations, governments, and its corporate and individual members to offer cloud security-specific research, education, certification, events and products. Cloud Security Alliance activities, knowledge and extensive network benefit the entire community impacted by cloud—from providers and customers, to governments, entrepreneurs and the assurance industry—and provide a forum through which diverse parties can work together to create and maintain a trusted cloud ecosystem.

CSA Address

709 Dupont St.
Bellingham, WA 98225, USA
Phone: +1.360.746.2689
Fax: +1.206.832.3513

Contact us: support@cloudsecurityalliance.org
Website: https://cloudsecurityalliance.org/
Zero Trust Training Page: https://knowledge.cloudsecurityalliance.org/page/zero-trust-training
Zero Trust Advancement Center: https://cloudsecurityalliance.org/zt/
Provide Feedback: support@cloudsecurityalliance.org
CSA Circle Online Community: https://circle.cloudsecurityalliance.org/
Twitter: https://twitter.com/cloudsa
LinkedIn: www.linkedin.com/company/cloud/security/alliance
Facebook: www.facebook.com/csacloudfiles
CSA CloudBytes Channel: http://www.csacloudbytes.com/
CSA Research Channel: https://www.brighttalk.com/channel/16947/
CSA Youtube Channel: https://csaurl.org/youtube
CSA Blog: https://cloudsecurityalliance.org/blog/

Acknowledgments

Dedicated to Juanita Koilpillai, a pioneer in software-defined perimeters whose contributions to the Zero Trust Architecture Training and CSA are immeasurable.

The Zero Trust Training was developed with the support of the Cloud Security Alliance Zero Trust Training (ZTT) Expert Group, whose members include volunteers from a wide variety of industries across the globe. Made up of subject matter experts with hands-on experience planning and implementing ZTT, both as cloud service consumers and providers, the ZTT Expert Group includes board members, the technical C-suite, as well as privacy, legal, internal audit, procurement, IT, security and development teams. From cumulative stakeholder input, the ZTT Expert Group established the value proposition, scope, learning objectives, and curriculum of the Zero Trust Training.

To learn more about the Zero Trust Training and ways to get involved please visit: https://cloudsecurityalliance.org/zt/

We would also like to thank our beta testers, who provided valuable feedback on the Zero Trust Training.

Lead Developers:

Daniele Catteddu, CTO, CISM, Cloud Security Alliance, Italy
Juanita Koilpilla, CEO, Waverly Labs, USA
Richard Lee, CISSP, CCSP, WCP, Citizens Financial Group, USA

Contributing Editors:

Anna Schorr, Training Program Manager, MBA, CCSK, Cloud Security Alliance, USA
Hannah Rock, Content Development Manager, Cloud Security Alliance, USA
James Lam, CISA, CISM, CRISC, CDPSE, TOGAF, M.S., Accenture Strategy & Consulting, USA
Jenna Morrison, CCSK, USA
Leon Yen, Technical Writer, Cloud Security Alliance, USA
Remo Hardeman, Security Architect, Cybersecurity Advisor, Omerta Information Security, Petro SA,
Stephen Smith, Graphic Designer, Cloud Security Alliance, USA

Expert Reviewer:

Anusha Vaidyanathan, USA
Juan Carlos (Charlie) Soto, MSc, CISSP, CISM, CDPSE, CIAM, CCISO, aCommerce, Thailand
Matthew Meersman, PhD, CISM, CISSP, CCSP, CDPSE, PMP, MITRE Corporation, USA
Michael J. Herndon, CCSP, CISSP, CRISC, CGEIT, CIPP/US, CIPT, AWS Certified Solution Architect, Bayer A.G., USA
Michael Roza, CPA, CISA, CIA, MBA, Exec MBA, CSA Research Fellow, Belgium
Nishanth Singarapu, CISM, CCSK, ZCEA, Neustar, USA
Robert D. Morris, CISSP, GDSA, GCIH, MITRE Corporation, USA
Ryan Bergsma, CCSK, Cloud Security Alliance, USA
Shamun Mahmud, Cloud Security Alliance, USA
Shinesa Cambric, CISSP, CISA, CCSP, CISM, Microsoft, USA
Vani Murthy, CISSP, CDPSE, CCSK, CRISC, PMP, ITIL, MBA, MS, Akamai Technologies, USA

Table of Contents

List of Figures

Course Intro

Welcome to your *Introduction to Software-Defined Perimeter* by Cloud Security Alliance. Please note that moving forward we will refer to Software-Defined Perimeter as SDP and to the Cloud Security Alliance as CSA. CSA is dedicated to defining and raising awareness of best practices to help ensure a secure cloud computing environment across the globe. We hope you are as excited to learn about SDP as we are about sharing this knowledge with you. This training module is part of a larger series of CSA programs on Zero Trust (ZT) that was created with the support of subject matter experts. If you are interested in volunteering with CSA to help our ongoing research efforts or are just interested in learning more about cloud security, please visit our website at cloudsecurityalliance.org.

This course is intended to give a high-level overview of why SDP was created, what it is, what it does, how it can be used, and how it relates to ZT and ZTA. Although it is not within the scope of this course to delve into SDP implementation how-tos, CSA will be releasing additional training courses that will elaborate on ZTA and further explore the details of SDP.[1]

Course Structure

This introductory course on SDP consists of four units, each geared towards helping learners gain competency in a specific area/topic:

- SDP History, Benefits, & Concepts
- Traditional Architecture Issues & SDP Solutions
- Core Tenets, Underlying Technologies, & Architecture
- The Basics of SDP Deployment Models

Course Learning Objectives

After completing this course, learners will be able to do the following:

- Explain what SDP is, how it came about, and what its technology and business benefits are
- Discuss the problems that SDP solves
- Describe some of SDP's underlying technologies
- Distinguish between the basic types of SDP deployments

[1] Cloud Security Alliance, "Zero Trust Architecture Training," https://cloudsecurityalliance.org/education/zero-trust-architecture-training

1 SDP History, Benefits, & Concepts

In this unit, you will be introduced to the concept of SDP, as well as gain a high-level overview of SDP architecture. Part of this introduction includes learning about the basics, such as the history of SDP, its technological and business benefits, as well as other related concepts.

1.1 SDP Definition & Function

CSA defines SDP[2] as a network security architecture implemented to provide security for all layers of the open systems interconnection (OSI) model. An SDP implementation hides assets and uses a single packet to establish trust via a separate control and data plane; only then are assets exposed to the requestor.

Although SDP has different roots than the ZT security model, the evolution of both concepts over time has led to community consensus in categorizing SDP as an implementation option of a ZTA. In order to isolate services from unsecured networks, SDP aims to give infrastructure and application owners the ability to deploy perimeter functionality when and where it's needed. SDP overlays existing physical infrastructure with logical components that should be operated under the control of the application owner. SDP only grants access to the application infrastructure after device attestation and identity verification.

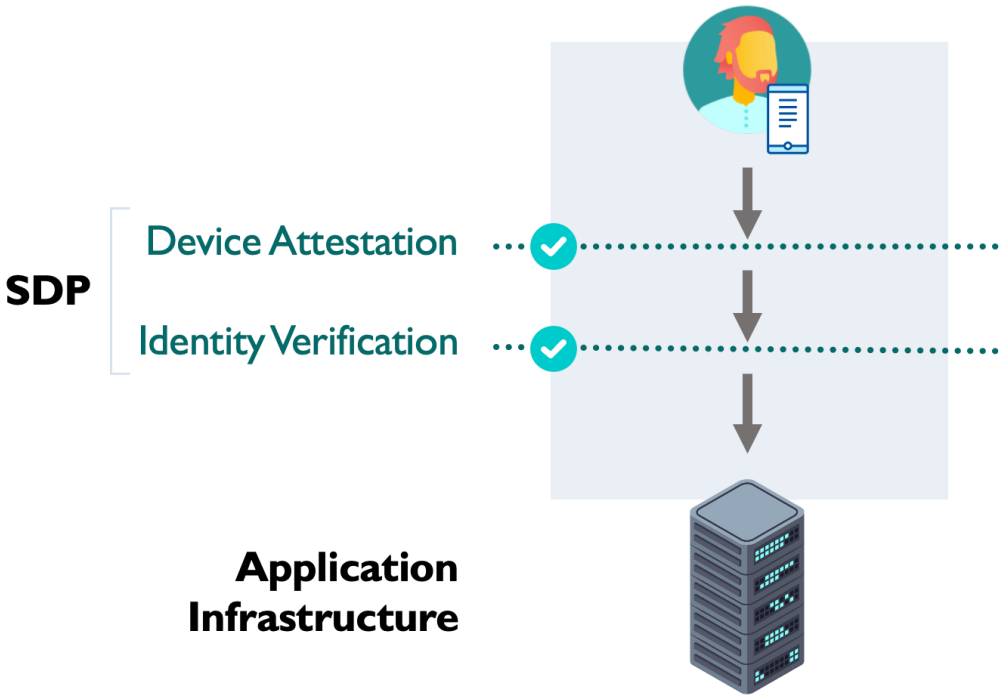

Figure 1: *Access Granted After Device Attestation/Identify Verification*

[2] https://cloudsecurityalliance.org/artifacts/software-defined-perimeter-and-zero-trust/

SDP is based on the premise that organizations should not implicitly trust anything inside or outside the network. It requires users on validated devices to cryptographically sign in to the perimeter created around hidden assets, even as they reside on public infrastructures. An SDP implementation hides assets with a drop-all firewall, uses a single packet to establish trust via a separate control plane, and provides mutual verification of connections in a data plane to hidden assets.

SDP brings together multiple controls that are usually separated by function and therefore hard to integrate: applications, firewalls, and clients, to name a few. These pieces of information need unification in order to establish and ensure secure connections. SDP helps to integrate controls for firewalls, encryption, identity and access management (IAM), session management, and device management into a comprehensive security architecture.

1.2 SDP Principles

The SDP architecture is based on the principles of least privilege and segregation of duties, enforced by implementing the following key controls and processes:

- Dynamic rules on drop-all firewalls
- Hiding servers and services
- Authentication before connections, for example not allowing connections before authorizing users on specific devices
- Using single packet authorization (SPA) and/or bi-directional encrypted communications like mutual transport layer security (mTLS)
- Fine-grained access control and device validation

1.3 Relationship Between SDP & ZT

In this section, you will learn about the relationship between SDP and ZT. ZT is the umbrella category under which SDP falls.

The ZT model is based on the following principles:

- Making no assumptions about the trustworthiness of an entity as it requests access to a resource
- Starting with no pre-established privileges, then relying on a construct which is used to add privileges
- Assuming breach and verifying all workforce, device, workload, network, and data access regardless of where, who, when, or to what resource

In essence, the ZT concept retires the use of trusted entities inside a defined corporate perimeter. Instead, it mandates that enterprises create micro-perimeters around sensitive data assets to maintain control and visibility around data use across the environment. Essentially, ZT aims to defend enterprise assets by distrusting anything inside or outside the perimeter. Implementing ZT requires verifying connection requests to assets before granting access, followed by continuous monitoring and evaluation throughout the entire duration. For additional references on ZT concepts

and architectures, please refer to the existing literature on the topic, and additional CSA training[3].

By comparing the foundational principles of SDP and ZT, it is clear that they are driven by the same high-level principle: "never trust, always verify". In fact, SDP is considered one implementation type of a ZTA; others include Zero Trust Network Access (ZTNA) and Google BeyondCorp, to name a few.

Compared to other ZTA implementations, SDP has some distinctive features and benefits, such as the use of a drop-all rule and the adoption of SPA. While these features are not necessarily unique to SDP, they are foundational to it; however, these features are not necessary requirements of other ZTA implementations.

> **NOTE:** SDP is a ZTA, but not every ZTA conforms with SDP requirements.

1.4 History of SDP

In this section, you will learn about the history of SDP, its origins, and why it was developed.

1.4.1 The Origination of SDP

SDP is a cybersecurity approach that evolved from the U.S. Defense Information Systems Agency's Global Information Grid Black Core Network initiative in 2007[4]. Designed to be extensible and future proof, this approach would later serve as the basis for CSA's SDP framework in 2013. The CSA SDP framework focuses on how to control access to resources based on identity and device attestation. Per SDP, connectivity is provided on a need to know model that verifies device posture and identity before granting access to an application infrastructure. Because the application infrastructure exists without visible domain name system (DNS) information or IP addresses, it is effectively hidden and undetectable unless access is specifically granted.

1.4.2 The Business Case for SDP

As organizations continue to undergo digital transformation, staying ahead of the threat landscape and attack chain curves is becoming increasingly difficult to achieve. Today, rather than managing and securing a single network, most organizations operate a variety of environment types, such as the following:

> - Physical, on-premises networks
> - Private clouds
> - Multiple public clouds
> - Virtual software-defined networking (SDN) environments

[3] Cloud Security Alliance, "Publications," https://cloudsecurityalliance.org/research/artifacts/
[4] DOD, "Vision for a Net-Centric, Service-Oriented DoD Enterprise," June 2007, https://www.acqnotes.com/Attachments/DoD%20GIG%20Architectural%20Vision,%20June%2007.pdf

Within these newer environments, organizations must facilitate the following:

- An expanding wide area network edge
- Information technology and operation technology convergence
- An increasingly mobile workforce

As organizations shift from traditional infrastructures to more virtualized and hybrid architectures, new attack vectors also emerge that require a novel approach to network security. SDP's designers focused on mitigating the most common network-based attacks, including server scanning, denial of service, SQL injection, operating system and application vulnerability exploits, man-in-the-middle, pass-the-hash, pass-the-ticket, to name a few. Despite the evolving cyber threat landscape, SDP continues to hold up against both existing and unknown threats.

1.5 Technology Benefits of SDP

In this section, we will explore the technological benefits of SDP. Some of these include SDP's attack surface reduction and pre-access authentication/authorization. We will also discuss SDP technological benefits such as IAM security and SDP's open specification.

1.5.1 Reduced Attack Surface

Today's network architectures consist of devices with assigned IP addresses used for connectivity. When a device is establishing a connection to another device, a handshake is established and authentication is verified. By reversing this sequence and first verifying the connection, SDP provides key technical benefits, most notably attack surface reduction. Connectivity to an organization's assets is provided only after authentication, validation/authorization, and the determination of which protected assets the user is allowed access to. These steps greatly reduce the attack surface of the application infrastructure.

With SDP, users and devices are no longer granted general access to network segments or subnets. Instead, policies ensure that users and devices only have access to specified hosts, resources, and/or services. Therefore, SDP can be used to protect different types of services or protocols such as Hypertext Transfer Protocol Secure (HTTPS) or remote desktop services (RDS). By controlling the access level that individual users and devices have to specific services, SDP can allow authorized users to access privileged services while hiding them from unauthorized users.

1.5.2 Authenticate & Authorize Before Access

SDP is an inherently comprehensive security architecture implemented using software components overlaid onto physical and virtual infrastructure. SDP uses a drop-all gateway to ensure that authentication and authorization is first performed in the control plane. By performing authentication prior to granting access to the perimeter, SDP ensures only users with appropriate authorization have access to the hidden infrastructure.

This functionality (i.e., providing connectivity to resources after authentication and authorization) is made possible by separating the control and data planes, providing enhanced protection by exposing assets only to verified users and devices. Fine-grained access control is implicit in SDP's design.

Without the SDP gateway's drop-all capability, allowing and enforcing only trusted connections would be prohibitively difficult. SDP's architecture enables pre-access vetting and fine-grained access policies through role and attribute-based permissions, as well as other similar access control mechanisms. Traditional architectures require separate implementations for each of these components, leading to increased complexity and higher maintenance overhead.

In contrast to IP-based alternatives, SDP provides a connection-based security architecture — this means access is granted per each independent connection, versus granting access to a device based on its allowlisted IP address.

SDP is a connection-oriented security architecture: while the physical infrastructure routes packets, SDP secures all connectivity over an infrastructure. This connection-based architecture distinction is important because of the current IP address explosion and the disintegrated perimeter in cloud environments — without SDP, IP-based security protections are ineffective when faced with this increasing complexity. SDP enables validation on the data plane prior to any Transmission Control Protocol/Transport Layer Security (TLS/TCP) handshake and enforces mutually encrypted communications. This practice helps to mitigate threats related to unauthorized access.

1.5.3 Centralized Organizational IAM Security

A prominent technical aspect of SDP is its centralized organizational IAM security. With IAM, a security problem on the front-end only requires an update to the SDP — every subsequent service within the perimeter will adjust to the heightened security measures. Traditional, direct access would require the checking and updating of potentially hundreds of services to address a single flaw. This is another example of how SDP drastically decreases maintenance overhead and complexity.

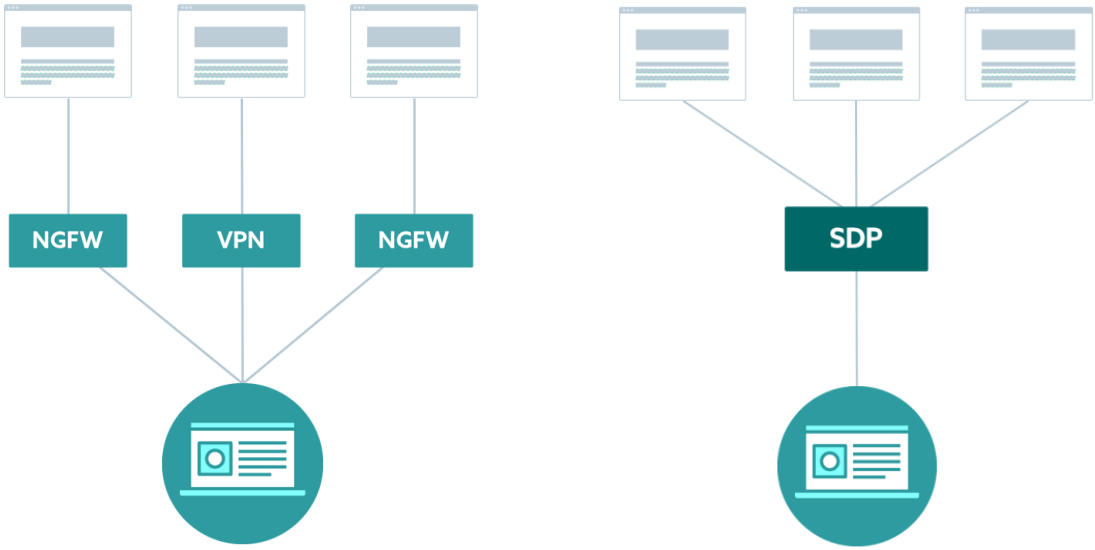

Figure 2: *Traditional IAM Security vs. SDP*

1.5.4 Open Specification

Open specifications are publicly available and therefore directly benefit from greater community contributions. This increases the volume of data flowing in, the validity, and practicality of the specification that is developed, based on a given set of data. With an open specification you can customize the code or implementation to your needs, audit the code as it exists, and receive community feedback on faults and errors.

The SDP specification is open and has been proven on many network implementations, such as SDNs, IoT networks, network functions virtualization, edge computing, 5G, and more. As part of the research efforts, the CSA Software-Defined Perimeter Working Group teamed up with the community at large to research how to create a high availability infrastructure using public clouds with the equivalent robustness of a dedicated data center. The CSA Software-Defined Perimeter Working Group has also created additional reference materials, such as *SDP Architecture Guide v2*[5] and *Software-Defined Perimeter as a DDoS Prevention Mechanism vs. SDP and DDoS*[6] that are publicly available. These documents were created with input from the global cybersecurity community.

1.6 Business Benefits of SDP

In this section, we will discuss the various business benefits that companies gain from implementing SDP. As part of this discussion, we will examine how SDP enhances existing cybersecurity investments, reduces costs and labor, and assists in governance, risk, and compliance (GRC) efforts.

1.6.1 Enhances Existing Cybersecurity Investments

Organizations are under continuous pressure to respond to security events in a timely manner; to this end, they've made substantial investments in cybersecurity. For example, expenditures in vulnerability management, patch management, and configuration management, have allowed organizations to lock down machines that utilize IP addresses for connectivity. Threat intelligence combined with endpoint threat detection and response (EDR) may also be in place, enabling organizations to better understand who the unauthorized users are and what connections they are making. Many organizations also manage their own security operation centers to actively monitor for threats and respond to intrusion alerts and other security events. SDP helps optimize security investments and make them more cost effective as a result of both preventive and reactive security capabilities.

SDP provides a preventive measure against network-based and cross-domain attacks. By hiding resources and applying the drop-all rules, SDP helps companies reduce their attack surface and consequently reduce the amount of security events or alerts that are collected by the security information and event management (SIEM) and sent to the security operation center. In addition, SDP reduces lateral movement in attacks by keeping assets invisible to unauthorized users. SDP helps reduce the

[5] Cloud Security Alliance, "SDP Architecture Guide v2," 7th, May 2019, https://cloudsecurityalliance.org/artifacts/sdp-architecture-guide-v2/

[6] Cloud Security Alliance, "Software-Defined Perimeter as a DDoS Prevention Mechanism," 27th, October 2019, https://cloudsecurityalliance.org/artifacts/software-defined-perimeter-as-a-ddos-prevention-mechanism/

complexity of integrating controls like firewalls, IAM, encryption, and device management by maintaining all rules in one place instead of addressing them for each individual implementation. This allows companies to focus internal resources on a smaller set of potentially negative events, therefore increasing the cost-effectiveness of the security investments.

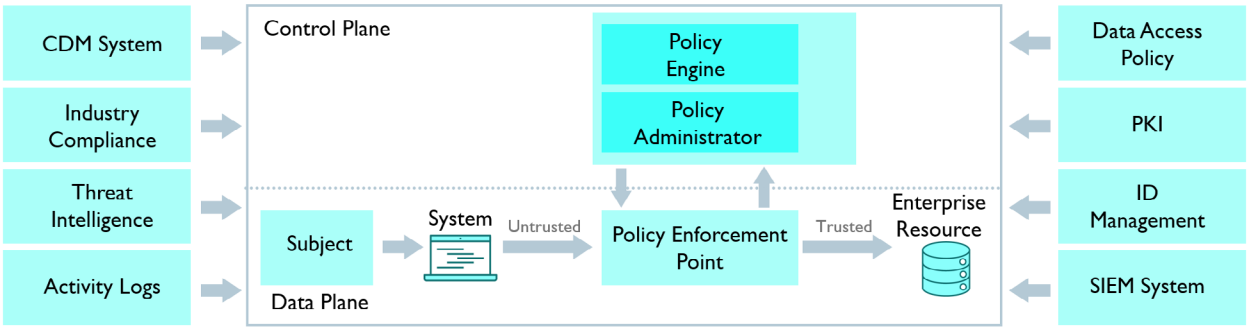

Figure 3: *SDP Ecosystem and Communication Flows*[7]

1.6.2 Cost Reduction & Labor Savings

Replacing traditional network security components with SDP reduces licensing and support costs. Implementing and enforcing security policies using SDP reduces operational complexity and reliance on traditional security tools. SDP also reduces costs of corporate backbone components by reducing or replacing multiprotocol label switching (MPLS) or leased line utilization. As information and communication technology environments change, reliance on corporate backbone is reduced, and more dynamic networks are implemented. SDP allows organizations to achieve dynamic network implementations securely. Ultimately, SDP brings efficiency and simplicity to organizations, which can ultimately help reduce scarce and often expensive labor needs.

1.6.3 Reduces Compliance Scope

As mentioned earlier, two of the main technology benefits of SDP are the reduction of the attack surface and an increased granular control over resource access. These two features, alongside micro-segmentation, are key to helping organizations better face compliance challenges, as they allow the reduction of the scope of compliance. By better controlling where regulated data are processed and stored, and by limiting, both physically and logically, who can have access to that data, organizations can reduce the scope of the compliance requirements. In addition, granular logging and monitoring of who-does-what-when-why support the creation of a much-needed accountability approach, which is foundational to any compliance effort.

[7] Figure adapted from NIST, "SP 800-207 Zero Trust Architecture," August 2020, https://csrc.nist.gov/publications/detail/sp/800-207/final

2 Traditional Architecture Issues & SDP Solutions

This unit reviews various issues that exist within current network security architectures. We will discuss how SDP protects against threats that exist due to those architectural inadequacies. In addition, we will explore how SDP integrates with industry adopted solutions or replaces them.

2.1 Concerns SDP Addresses

In this section you will learn about critical issues that SDP addresses, including the changing perimeter, the IP address challenge, and the integration of security controls.

2.1.1 The Shifting Perimeter

Virtualized networks have superseded the older, fixed network perimeter paradigm that relies on trusted internal network segments protected by network appliances (e.g., load balancers and firewalls). Network protocols of the past are not secure by design and are known to have vulnerabilities. In addition, the plethora of mobile and IoT devices further challenge the validity of a fixed network perimeter.

The introduction of the cloud has drastically changed the composition of organizations' IT environments. With the emergence of bring your own device (BYOD), machine-to-machine connectivity, the rise in remote access, and phishing attacks, legacy security approaches are no longer effective in protecting the shifting physical perimeter. For one, there are more internal devices and varieties of users. For example, contractors working on-site may require temporary access to IT resources, both on-premises and in the cloud. IT environments are also increasingly diversified with the continued enterprise adoption of hybrid architectures. Corporate devices are moving to the cloud, co-located facilities, and in some cases to off-site customer and partner facilities. These migrations further shift the physical perimeter of the organization; SDP addresses the inherent challenges of securing this shifting physical perimeter with a software overlay that creates virtual perimeters dynamically, when and where they are needed.

2.1.2 The IP Address Challenge

Everything on the internet today relies on TCP/IP for trust. This is problematic, because IP addresses have no concept of users' identities. TCP/IP simply addresses connectivity — it doesn't validate the endpoint or the user as being trustworthy.

TCP/IP is a bidirectional protocol, so internal trusted hosts communicating with external untrusted hosts can receive unsafe messages. Any changes to IP addresses may require extensive reconfiguration resulting in potential security group and network access control (NAC) list errors. Unmanaged/forgotten internal hosts can provide an entry point for malicious actors by providing default responses using legacy protocols such as ICMP. This illustrates that common use of network address translation (NAT) tables combined with TCP/IP is inherently open to compromise.

IP addresses should not be used as anchors for network locations because they are location-dependent (i.e., users' devices are assigned new IP addresses when they are relocated). SDP tackles this IP address challenge by securing connections while being IP address agnostic. This means that the SDP is aware of IP addresses but doesn't rely on them for authorizing access to protected resources.

2.1.3 Integrating Security Controls

The integration of multiple security controls like firewalls and identity managers is typically implemented to achieve compliance. However, integrating these controls to work as a whole in protecting the application infrastructure can be challenging. Currently, the integration of controls may be performed by gathering data in an SIEM for analysis; however, correlating disparate streams of security to gain deeper insights (e.g., who is connected, from what device, from where, to what, and more) is resource intensive.

A single point of trust for network connections requires the following:

> - Information about users, provided by the applications
> - Information about the network, provided by firewalls
> - Information about devices, provided by the client

These disparate requirements make it difficult to implement an integrated set of controls for a physical network. Furthermore, integrating identity management prior to allowing access through a firewall requires the routing of packets to a different service — one that is resource-intensive and may or may not be proxied. In addition, most DevOps teams consider application layer firewalls and anti-denial of service/distributed denial of service (DoS/DDoS) protection as an afterthought; moreover, allowing individual applications to control their own security posture may result in catastrophe. Integrating access control, identity management, session management, and firewall management in today's environments is highly difficult; SDP addressed this challenge by providing a unified location for implementing and managing controls for the entire environment, versus using traditional distributed controls.

2.2 Threats SDP Protects Against

In this section, we will analyze the efficacy of SDP for reducing cyber risk and mitigating threats. We will present well-known threats/cyber risks published by the OWASP, Verizon, and CSA that demonstrate the real value of ZTA using the SDP. As illustrated below, the integration of SPA and SDP with enterprise IAM helps raise the bar for security. The tables provide a high-level overview of the relevant risks/threats, results of a successful exploit execution, and how SDP can be leveraged to prevent these security incidents from occurring.

2.2.1 CSA's Egregious 11

Risk/Threat	Result(s) of Successful Exploit Execution	SDP Mitigation
Data breaches	Reputational damage, loss of customer/partner trust, loss of intellectual property to competitors which may impact product releases, regulatory implications that may result in monetary loss, brand damage/ market value loss, legal and contractual liabilities, and financial expenses incurred due to incident response and forensic analysis	SDP has a drop-all firewall that drops packets not explicitly configured, preventing data breaches and/or preventing their scope of damage.
Misconfigurations & inadequate change control	Exposure of data stored in cloud repositories	SDP assists in change control by providing access configured for changes only after approval.
Lack of cloud security architecture & strategy	Financial loss, reputational damage, legal repercussions, and fines	SDP has a ZT policy that outlines a framework with systems designed around the value of the data and its specific protection needs.
Insufficient identity, credential, access, & key management	Unauthorized access, exfiltration, modification, deletion of data, issuing of control plane and management functions, eavesdropping on data in transit, and the release of malicious software that appears to originate from a legitimate source	Authentication, authorization, and mutual factor authorization (MFA) is at the core of SDP; subsequently, using SDP integrated with enterprise and cloud IAM/ identity provider (IdP) reduces the attack surface.

Account hijacking	Complete deletion of organization assets, data and capabilities, data leaks and resulting brand/reputational damage, legal liability due to sensitive personal and business information exposure	Authentication, authorization, and MFA is at the core of SDP; using SDP integrated with enterprise and cloud IAM/ IdP limits the exposure for account hijacking.
Insider threat	Loss of proprietary information and intellectual property, system downtime impacting company productivity, and other customer data losses that reduce their confidence in the organization's services	SDP includes micro-segmentation of the organizational environment to ensure that access to resources are granted on a need to know basis. SDP's continuous logging integrated with user entity behavior analytics can limit the data loss and/or alert on malicious/ abnormal activity and behavior.
Insecure interfaces & APIs	Regulatory and financial impact in the form of fines/penalties, security issues related to confidentiality, integrity, availability and accountability	SDP provides controls defining communication endpoints (as long as the interface and API communications sit behind the SDP controller).
Weak control plane	Data loss, either due to theft or corruption, resulting in a substantial impact on the business — particularly if the incident includes privileged user data, and regulatory punishment for data loss may be incurred	SDP's control plane is protected by both network level controls (e.g., SPA), and strong authentication.
Metastructure/application infrastructure failures	Failures at the cloud service provider level, resulting in customers being severely impacted, and tenant misconfigurations could result in financial losses and operational disruptions	SDP limits the impact of misconfigurations by hiding resources behind the gateway/ controller.

Limited cloud usage visibility	Lack of governance, awareness, and security	SDP logs all inbound activity, providing better visibility and situational awareness.
Abuse of cloud services	Financial losses due to excessive metered cloud use (e.g., attackers using compromised cloud servers as a malware distribution host)	SDP safeguards access to stateful (e.g., security group/ network security group) and stateless (e.g., access control list/network access list) firewall configurations. Coupling security group/ network security group and access control list/ network access control list configurations enables the dropping of unauthorized traffic (e.g., for mining cryptocurrency or distributing malware).

Table 1: Top Threats to Cloud Computing: Egregious Eleven Deep Dive[8]

2.2.2 Verizon's DBIR

Risk/Threat	Result(s) of Successful Exploit Execution	SDP Mitigation
Phishing & social engineering	Acquisition of credentials	SDP's integration with domain-based, message authentication/reporting, as well as its requirements for validating source networks and capabilities (e.g.,MFA and device fingerprinting) reduce the risk of these attacks.

[8] Cloud Security Alliance, "Top Threats to Cloud Computing: Egregious Eleven Deep Dive," 23rd, September 2020, https://cloudsecurityalliance.org/artifacts/top-threats-egregious-11-deep-dive/

Web application attacks	Stolen credentials and successful brute force attempts that enable unauthorized access to web application servers, mail servers, and others IT assets, resulting in compromised privileged data (e.g., medical records, employee data)	SDP's use of SPA for inbound/outbound server traffic coupled with MFA helps prevent these attack types (i.e., requiring authentication prior to authorizing access).
Lost or stolen credentials	Exposure of sensitive data	SDP's MFA requirement minimizes the impact of stolen credentials, since malicious actors are not given explicit access to resources.
Ransomware	Revenue loss and supply chain disruption	SDP prevents the installation of unapproved software and potentially malicious applications (e.g., ransomware) on servers.
Miscellaneous errors compromising security	Eavesdropping, data loss, data exposure, and unauthorized access	SDP requires authentication prior to accessing applications and/or server resources.
DoS	Loss of service and/or service disruption	SDP controls communication endpoints and is therefore stateless; drop-all firewalls block threats such as malware and command and control servers.
System intrusion	Eavesdropping, data loss, data exposure, and unauthorized access	SDP requires the use of SPA to/from the server. Coupled with MFA, these controls help to enforce authentication prior to authorized access.
Privilege abuse	Eavesdropping, data loss, data exposure, and unauthorized access	SDP's MFA requirement prevents unauthorized access and escalation of privileges from occurring.

Table 2: DBIR- 2021 Data Breach Investigations Report[9]

[9] Verizon, "2021 Data Breach Investigations Report," 2021, https://enterprise.verizon.com/resources/reports/2021-data-breach-investigations-report.pdf

2.2.3 OWASP IoT Top 10

Risk/Threat	Result(s) of Successful Exploit Execution	SDP Mitigation
Weak, guessable, or hard coded passwords	Unauthorized access	SDP authenticates users prior to granting them access; additionally, MFA helps mitigate the risk of stolen/lost credentials and devices.
Insecure network services	Unauthorized access	SDP requires encryption for enforcing confidentiality in an insecure network. For example, devices must support encryption in order for SPA over HOTP (HMAC One Time Password) and data communications via mTLS to function (in protecting confidentiality).
Insecure ecosystem interfaces	Unauthorized access	SDP unifies the different ecosystem interfaces into a secure, single source of truth.
Lack of secure update mechanism	Unauthorized access	SDP and SPA provide device authentication and valid endpoints via mTLS, allowing for secure over-the-air authentication and device update mechanisms.
Use of insecure or outdated components	Eavesdropping, data loss/ exposure, and unauthorized access	SDP leverages ZT call flows in the TCP/IP network, thereby protecting legacy, insecure or outdated components.
Insufficient privacy protection	Eavesdropping and data loss/ exposure	SDP requires encryption in order for SPA over HOTP to function and ensure confidentiality.

Insecure data transfer & storage	Eavesdropping and data loss/ exposure	SDP requires encryption in order for SPA over HOTP and data communications via mTLS to function and ensure confidentiality.
Lack of device management	Unauthorized access	SDP provides secure mobile device management by enabling device management and software updates via SPA and mTLS.
Insecure default settings	Unauthorized access	SDP requires micro-segmentation as well as MFA to mitigate the risk of outdated/unpatched and misconfigured devices.
Lack of physical hardening	Unauthorized access	SDP couples automated device auditing with secure device management to validate device security postures.

Table 3: OWASP IoT Top 10[10]

2.2.4 OWASP Top 10

Risk/Threat	Result(s) of Successful Exploit Execution	SDP Mitigation
Broken access control	Unauthorized access	SDP authenticates users and validates that requests are authorized prior to granting access.
Cryptographic failures	Exposure of sensitive data or a compromised system	SDP enforces cryptography requirements (e.g., in mTLS sessions).
Injection	Malicious injection and alteration of responses to compromised application server	SDP mitigates application attacks through its inherent MFA and SPA/drop-all approach.

[10] OWASP, "OWASP IoT Top 10," 2018, https://owasp.org/www-pdf-archive/OWASP-IoT-Top-10-2018-final.pdf

Insecure design	Exploitation of vulnerabilities	SDP helps bolster threat modeling, secure design principles, patterns, and design practices affecting reference architectures and configuration audits.
Security misconfiguration	Exposure of data and exploitation of application vulnerabilities	SDP mandates micro-segmentation and MFA — critical SDP features for mitigating the impact of security misconfigurations. MFA limits the escalation of privileges and reduces the blast radius of attacks.
Vulnerable & outdated components	Exploitation of known vulnerabilities	SDP prevents legacy, insecure, or outdated components from being attacked by hiding the associated services from unauthorized users/devices.
Identification & authentication failures	Privileged access escalation and lateral movement	SDP mandates micro-segmentation and the granting of access to resources based on the requester's need to know/ need for access.
Software & data integrity failures	Insertion of malicious code into critical path continuous integration/continuous delivery (CI/CD) pipelines (e.g., open source software, containing malicious code)	Leveraging the SPA, SDP uses endpoint authentication to assist with verifying the integrity of CI/CD pipelines and software updates.
Security logging & monitoring failures	Lack of visibility into unauthorized or malicious events	SDP enforces comprehensive and continuous monitoring. With logging/monitoring services in place per SDP's requirements, security incidents can be remediated in a timely manner.

Server side request forgery	Unauthorized access and compromise of vulnerable applications and related/ connected back-end systems. Attackers may also use this exploit method to circumvent user input validation	SDP helps mitigate attacks to applications exposed on a network through its inherent MFA and SPA/drop-all approach.

Table 4: 2021 Draft OWASP Top 10[11]

In addition, the following sections address some of the various threats that SDP helps protect against. These include server exploitation and hijacking, among others.

2.2.5 Server Exploitation Threats

SDP features like server isolation, SPA, and dynamic drop-all firewalls bolster application infrastructure security and help protect against server exploitation threats such as the following:

- DoS/DDoS attacks
- Code injection attacks
- Other attacks that exploit server misconfigurations/vulnerabilities

2.2.6 Hijacking Threats

SDP attributes such as encryption, pinned certificates, and non-reliance on DNS protect against connection hijacking threats like the following:

- Man-in-the-middle (MITM) attacks
- Certificate forgery
- DNS poisoning
- Code injections

2.2.7 Other Threats

SDP features like MFA, mTLS, and device fingerprinting protect against the following:

- Phishing
- Keyloggers
- Brute force attacks

For further information, please refer to CSA's *SDP Architecture Guide*[12] and existing research on SDP and ZT[13].

[11] Footnote 12: OWASP, "OWASP Top 10," 2021, https://owasp.org/Top10/
[12] Cloud Security Alliance, "SDP Architecture Guide v2," 7th, May 2019, https://cloudsecurityalliance.org/artifacts/sdp-architecture-guide-v2/
[13] Cloud Security Alliance, "Software-Defined Perimeter (SDP) and Zero Trust," 27th, May 2020, https://cloudsecurityalliance.org/artifacts/software-defined-perimeter-and-zero-trust/

2.3 SDP & Industry Adopted Solutions

In this section, you will learn about various industry adopted solutions and how SDP replaces or works in conjunction with them. This includes NAC, virtual private networks (VPNs), IAM, and next generation firewalls (NGFW).

2.3.1 Network Access Control

NAC typically controls what devices can connect to a given network and which network locations or segments they have access to. These solutions use a combination of standards-based hardware (e.g, 802.1X for port-based NAC) and software to validate devices, prior to granting them network access. NAC typically operates at layer 2 (i.e., the data link layer) of the OSI model.

When a device first appears on the network, the NAC performs device validation followed by assignment to the correct network segment (e.g., virtual local area network). In practice, NACs coarsely assign devices to a small number of networks, as most organizations only have a few networks set up (e.g., guest, employee, and production). Because NACs operate at layer 2 of the OSI model, they more often require specific network equipment, don't operate in cloud environments, and are not used by remote users.

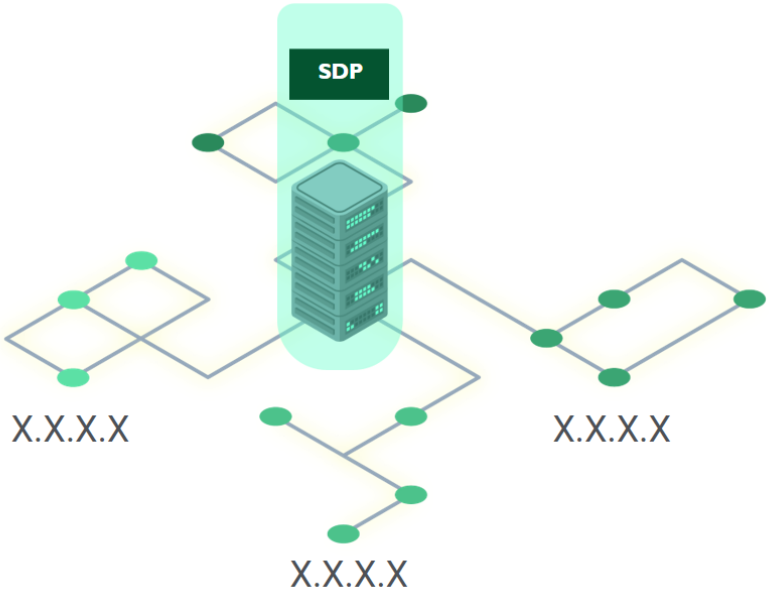

Figure 4: *SDP as NAC Replacement*

In some respects, SDP can be considered a modern replacement for NAC. Though they share similar functionalities, SDP, unlike NAC, does not require specific network hardware to function. This allows for the integration of users and provisioning of device access without a dedicated network appliance. SDP fully supports cloud environments and remote access, overcoming traditional NAC limitations. However, some environments are more suitable for NAC implementations — for example, those with printers, copiers, landline phones, or security cameras. These devices are often 802.1X compliant with built-in support, which means they don't typically support the installation of an SDP client. In this case, the gateway-to-gateway model is a better option for protecting and managing access to these devices.

2.3.2 Virtual Private Network

VPNs establish secure private network connections over untrusted networks. Commonly used for secure remote access (e.g. employee access to a corporate site, secure site-to-site communications, or site-to-site extranets between companies), VPNs use TLS/Secure Sockets Layer (SSL) or Internet Protocol Security (IPSec) to establish an encrypted tunnel.

Although VPNs encapsulate and encrypt network traffic, they also allow unrestricted access to a network segment. This is risky, especially if credentials are compromised. In contrast, SDP will only allow access to specifically assigned applications in the network segments.

On the user experience side, VPNs tend to impose a considerable burden on users, especially in environments undergoing significant cloud-based transformations and migrations. IT may also need to configure VPN for users requiring secure access to multiple sites, as this prevents unintentional network bridging and systems from connecting to multiple locations simultaneously. Ultimately, this shifts the burden and inconvenience of switching back and forth between remote locations on the user.

1. In distributed environments, VPNs may require the unnecessary backhauling of user traffic through a corporate data center, adding latency and bandwidth costs.
2. VPN servers themselves expose the network on the internet. VPN servers contain security vulnerabilities as do most IT components, which an attacker could exploit to gain access and exfiltrate data or perform other malicious activities.
3. VPN licensing costs are not expensive, but anecdotally they can be difficult to implement and maintain. Whenever cloud migration is involved, VPN management balloons in complexity. This is because IT administrators need to configure and sync VPN and firewall policies across multiple locations,making it even more difficult to mitigate unauthorized access.

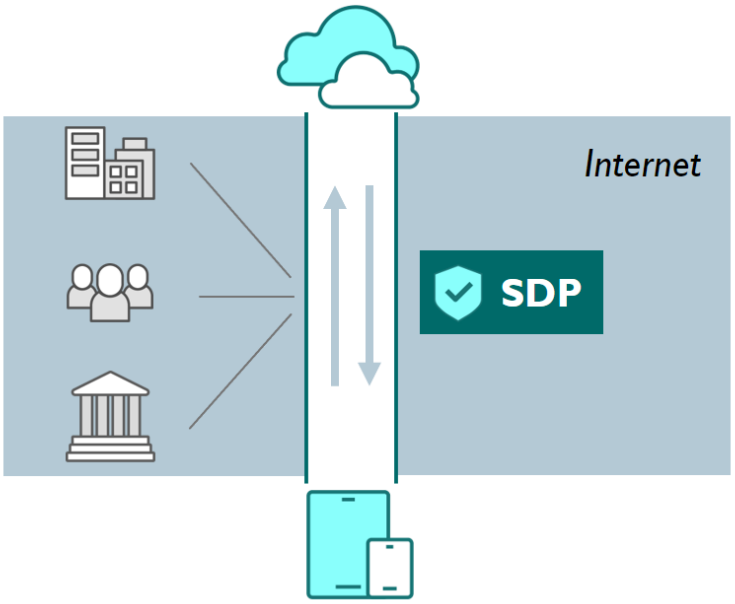

Figure 5: *SDP as VPN Replacement*

VPNs are a prime technology use case for replacement by SDP — however, it's worth noting that SDP can work alongside existing VPNs or replace them entirely, depending on the deployment model. However, both require an installation of a client on the user's device. By using SDP instead of VPNs, organizations can have a single access control platform consistent for secure access to cloud, remote, on-premises, and mobile device users. Since SDPs enable zero visibility via SPA and dynamic firewalls, they are considerably more resilient to cyber attacks than traditional VPN servers.

2.3.3 Identity & Access Management

The SDP architecture is designed to integrate with existing enterprise IAM providers in the cloud, on-premises, or hybrid environments. IAM provides a unified mechanism for users and devices to be validated, authenticated, and authorized. It provides a way to store managed identity attributes and group memberships within a central system using protocols to enable access directly or via federation. SDP supports standard protocols and security mechanisms used by IAM, including Lightweight Directory Access Protocol (LDAP), Active Directory (AD), and Security Assertion Markup Language (SAML).

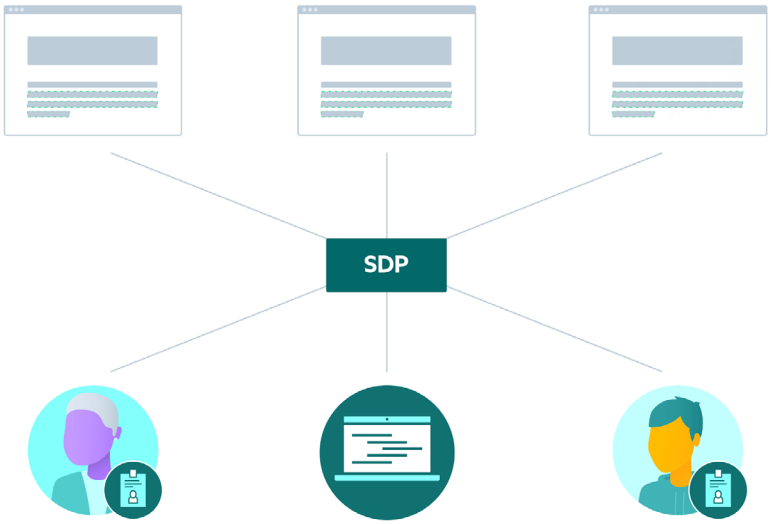

Figure 6: *SDP and IAM*

SDP typically controls access based on business rules. These rules can be built up from IAM attributes and group memberships, as well as from the attributes of devices making the connection and/or the network segments themselves. The telemetry data provided by these sources enables the creation of granular access rules for allowing/restricting access. This ensures only users requesting specific access on registered devices are granted authorization to the resources in question.

Integration of SDP with IAM is not only used for initial user authentication; it's also commonly used in conjunction with step-up authentication (e.g., prompting for a one-time password to access sensitive resources). IAM systems can also communicate with an SDP via API calls, as SDP can respond to identity lifecycle processes in this configuration. Some examples include joiners, mover, and leavers (JML), disabling an account, group membership changes, and dropping user/device connections from certain geographic locations.

In order to authenticate users, SDP must leverage IAM to access identity telemetry information that the SDP controller uses to make authorization decisions. IAM data is not only used to augment the SDP controller's capabilities — it's also used for populating audit logs with additional details regarding user and device access (e.g., access granted/denied details). Compared to traditional network access and IP address information, IAM telemetry correlates application access to users, yielding far more useful data with less overhead. This reduced overhead is leveraged primarily by IT when auditing historical access records in security or compliance use cases.

2.3.3.1 SDP & Identity Lifecycle Management

In identity lifecycle management, IAM tools focus on the business processes for maintaining the identity lifecycle (i.e., the JML process). IAM standardizes how identity information is used to control access to resources, using access methods such as role-based and attribute-based access control.

SDP supports these IAM processes and relies heavily on IAM-managed identity attributes and group memberships. As user attributes or group memberships change, SDP will alter access permissions accordingly without changing IAM telemetry, as SDP is a downstream system. These processes are utilized by SDP via standard protocols like SAML, AD, LDAP, or through the use of APIs.

2.3.3.2 SDP & Open Authentication Protocols

SDP integrates with open authentication protocols such as SAML. Within an SDP deployment, a SAML entity might act as an identity provider for user attributes and/or as an MFA authentication provider.

In addition to SAML, SDP integrates with many other open authentication protocols such as OAuth, OpenID Connect, W3C Web Authentication, and the FIDO Alliance Client-to-Authenticator Protocol. These protocols will be explored in future SDP-related research, but are not in scope for this training.

2.3.4 Next Generation Firewall

NGFWs have all the capabilities of traditional firewalls, along with additional capabilities such as intrusion detection/prevention and deep packet inspection. NGFWs filter data using the information in layers 2 through 4 of the OSI model (i.e., the data-link, network, and transport layers). Additionally, NGFWs use the information in layers 5 through 7 (i.e., the session, presentation, and application layer) to perform additional functions.

Depending on the vendor, NGFWs may provide some or all of the following capabilities:

- Application awareness – recognizes applications to determine what attacks to look for
- Intrusion detection/prevention system (IDPS) – monitors the security status of the network and denies traffic to prevent security problems
- Identity awareness (user and group control) – controls which resources users can access
- VPN – allows for remote user access across an untrusted network

While NGFWs represent a significant improvement over traditional firewalls, they still have their limitations. Some of these include:

- Latency – as is the case with IDPS, NGFWs will cause additional network latency, especially if they're performing file inspection
- Scalability issues – a NGFW requires more robust hardware to scale
- Rule complexity – some NGFW vendors include identity management capabilities such as user/group attribute assignments, but anecdotally these tend to be complex to implement

SDP is a natural complement to existing NGFWs. Enterprises can use SDP for secure user access policies while leveraging their NGFWs for core firewall, IDPS, and traffic inspection capabilities. By integrating SDP with a NGFW, enterprises can at once enforce the zero visibility principle and make them more dynamic.

User access policies can be achieved by integrating NGFWs with IAM or AD. By combining NGFW VPN capabilities with user and application awareness, enterprises can, to some degree, accomplish many of the goals of SDP. However, there are some general architectural differences.

NGFW systems are IP-based and offer limited identity and application-centric capabilities, whereas SDP is connection-based and therefore easier to control authorized connections. Additionally, NGFWs tend to be much less dynamic than SDPs, while the latter often supports the ability to include external systems in access decisions. For example, a prime use case for SDP is to only permit developer access to staging servers during an approved change management window.

Since NGFWs are still firewalls, their network deployment/design patterns still favor traditional perimeter-centric network architectures with site-to-site connections between locations. On the other hand, SDP deployments usually support more distributed and flexible networks, thereby enabling a flexible network segmentation capability.

SDP is fundamentally based on a need to know security principle, which by design hides all unauthorized services from users and leverages SPA and dynamic firewalls to hide connections protected by the SDP. NGFWs are not designed to function this way and typically result in environments that are more visible and therefore higher risk than with SDP. It should be noted that NGFWs have not yet been able to integrate authentication and authorization controls prior to allowing connections.

3 Core Tenets, Underlying Technologies, & Architecture

In this unit you will learn the foundation of how SDP works and how it accomplishes its task of providing network security. We will explain SDP's core tenets, underlying technologies, architectural components, and secure workflow.

3.1 SDP Core Tenets

SDP has three core tenets that govern its implementations: assume nothing, trust no one or thing, and validate everything. These tenets are used as the building blocks of the SDP framework. SDP was designed to secure dynamic workloads, most prominent in cloud mobile environments, by providing the following:

- Software-defined, dynamic, endpoint validation
- A connection-based paradigm
- Integration of firewalls, identity and access, session, encryption, and device management

These design features are covered in CSA's *SDP Specification v2*[14].

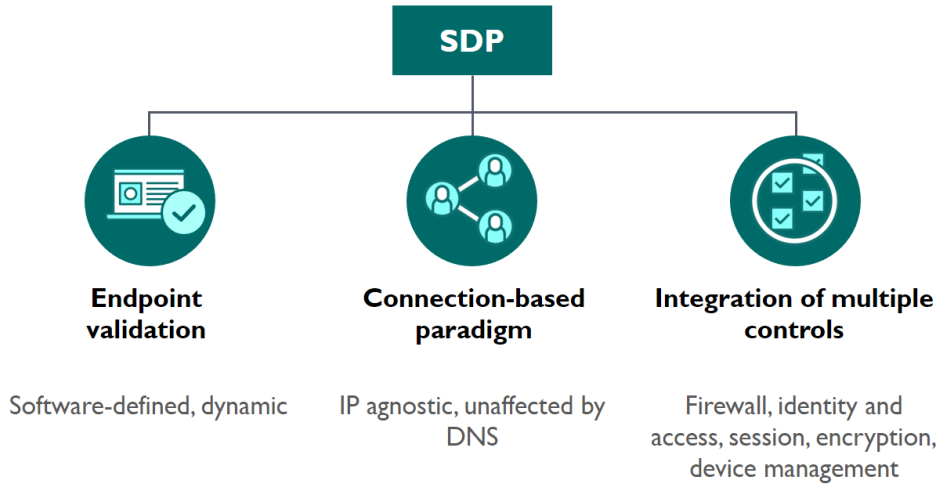

Figure 7: *SDP Core Tenets Tree*

[14] Figure adapted from Cloud Security Allaince, "Software-Defined Perimeter (SDP) Specification v2," 10th, March, 2022, https://cloudsecurityalliance.org/artifacts/software-defined-perimeter-zero-trust-specification-v2/

3.2 Underlying Technology

In this section, we will introduce and discuss the underlying technologies that support the SDP architecture, including drop-all firewalls, separate control and data planes, mTLS, and SPA. SDP provides a security architecture designed from the ground up using these foundational technologies.

3.2.1 Drop-All Firewall

The drop-all firewall is the most critical underlying technology of the SDP. Using drop-all rules, these firewalls operate according to the principle of least privilege: all actions not explicitly allowed remain forbidden. This strategy supports the ability to add rules to the firewall dynamically for post-authentication access level changes.

An SDP deployment can identify and deny risky transactions based on the analysis of a single packet. When malicious actors attempt to connect, SDP uses a drop-all rule to drop all unauthorized packets at the perimeter. This approach is highly effective as it only focuses on allowing approved actions, rather than blocking unapproved actions.

3.2.2 Separate Control & Data Planes

The three basic components of an SDP architecture are the data plane, control plane, and management plane. The data plane — also known as the user plane, forwarding plane, carrier plane, or bearer plane — is the part of a network that carries user traffic. In SDP, the control plane and management plane enable the data plane, which bears the traffic that the network carries. The control plane is responsible for establishing connections and dropping unauthorized packets at the perimeter. The control plane takes care of authenticating and authorizing users and devices prior to sending data to the SDP gateway. No devices are allowed to reach the data plane until the user and device in question are validated at the control plane.

In traditional architectures, data and control planes are commonly implemented together. In contrast, SDP architectures place the control plane outside the organization's perimeter; subsequently users and devices do not enter the organization's environment until they are authenticated and authorized. By separating the data plane from the control plane, SDP enables the external control plane to perform authentication and authorization before granting access to resources.

3.2.3 Mutual Transport Layer Security

SDP uses mTLS authentication to ensure that client-server traffic is secure and trusted in both directions. This allows requests that do not log in with an identity provider (e.g., requests from IoT devices) to demonstrate that they are permitted access to a given resource. Client certificate authentication adds an additional security layer for team members who both log in with an identity provider and use a valid client certificate for authentication.

Chiefly, mTLS is ideal for use in the following IT environments:

- A limited number of programmatic and homogeneous clients connect to specific web services
- The operational burden is limited
- Security requirements are more stringent compared to consumer environments

Subsequently, mTLS authentication is more widely used in business-to-business applications.

3.2.4 Single Packet Authorization

SPA is a protocol that allows a user to make a request to a server. This request cannot be replayed and uniquely identifies the user. SDP uses SPA to compensate for the fundamentally open and insecure nature of TCP/IP. In addition, SDP uses SPA to authorize a valid device and authenticate a user identity. SPA then permits access into the perimeter and the relevant system component. The purpose of SPA is to allow assets within the perimeter to be restricted via a default drop-all firewall.

While implementations of SPA may differ slightly, they should share the following common concepts for an SDP implementation:

- An SPA packet must be encrypted and authenticated
- An SPA packet must self-contain all the necessary information
- Packet headers are not considered trustworthy
- A SPA packet must not depend on administrator or root level access in order to generate and send
- There is no raw packet manipulation
- The server must receive and process the SPA packet as silently as possible, no response or verification is sent

3.2.4.1 SPA Benefits

The key advantage of using SPA is service restriction. A default drop-all firewall posture prevents port scanning and other attacker-related reconnaissance techniques. It effectively renders the SPA components invisible to unauthorized users, significantly reducing the attack surface of the SDP system. This compares favorably to systems such as VPNs, with open ports and known vulnerabilities in many implementations.

There are subsequent benefits to restricting services. One is zero-day protection. Any newly discovered vulnerability becomes significantly less critical when only authenticated users can access the affected service. Another benefit is DDoS protection. A relatively small amount of traffic can take an HTTPS service offline if that service is exposed to the public internet for attack. A SPA makes that service visible only to authenticated users. Therefore, a DDoS attack is handled by a default drop-all firewall instead of the protected service itself.

One of the core goals of SDP is to overcome the fundamentally open, or insecure nature of TCP/IP, which follows a connect, then authenticate model. Amid today's threat landscape, it's simply

unacceptable to permit malicious actors to scan and connect to enterprise systems. There are far too many known and unknown vulnerabilities in systems to allow this. SPA and SDP solve this problem in two ways. First, applications using the SDP architecture are hidden behind an SDP gateway so that they're only accessible to authorized users. Second, the SDP components themselves, the controller and gateway, are protected by SPA. This allows them to be securely deployed with internet-facing placement, ensuring that legitimate users have productive and reliable access, while they remain invisible to unauthorized users.

3.2.4.2 SPA Limitations

SPA is only a part of SDP and is not a complete security architecture on its own. While SPA implementations should be designed to be resilient to replay attacks, SPA may be subject to a MITM attack; specifically, if MITM adversaries are able to capture or alter the SPA packet, they can potentially establish the TCP connection to the controller or accepting host (AH) in place of the authorized initiating host (IH). However, these adversaries will be unable to complete the mTLS connection, since it will not have the client's certificate. The controller or AH should therefore reject this connection attempt and close the TCP connection. Even considering this limitation, which only applies to the MITM scenario, SPA is more secure than standard TCP.

3.3 SDP Architecture Components

In this section, we will discuss the foundational SDP architecture components: IH, AH, gateways, SDP clients, and the controller. SDP provides an integrated security architecture that is otherwise hard to achieve with security point products.

SDP integrates the following discrete architectural elements:

- Identity-aware applications
- Client-aware devices
- Network-aware firewalls/gateways

3.3.1 Initiating Hosts

IH initiate connections to the SDP. IH are devices, including laptops, tablets, and smartphones that SDP client software is run on. This host environment may be on a network outside the control of the enterprise operating the SDP.

3.3.2 SDP Client

The SDP client consists of software installed on the IH device. The client initiates connections in order to cryptographically sign in to the SDP. The SDP client typically generates the SPA packet for the SDP gateway after completing the authentication and authorization process with the SDP controller.

3.3.3 Accepting Hosts

AH are devices that accept connections from IH and provide a set of services that are protected by the SDP. They typically reside on a network under the control of the enterprise (and/or direct representative) operating the SDP, and do not acknowledge communications from any other host or respond to non-provisioned requests. To unauthorized users and devices, AH remain cloaked and inaccessible while using SDP's SPA.

3.3.4 Controller

The SDP controller is an appliance or process that secures access to isolated services. It does this by ensuring that users are authenticated and authorized, devices are validated, secure communications are established, and user and management traffic on a network remain separate. Like the AH, the controller is also protected by SPA, making it invisible and inaccessible to unauthorized users and devices. Both IH and AH connect to the SDP controller.

3.3.5 Gateway

The SDP gateway is an appliance or process that provides access through the invisible perimeter for authorized users and devices. Through this gateway, authorized users and devices are able to access protected processes and services. The gateway can also effectively allow monitoring, logging, and reporting on these connections. The functionality of the gateway depends on where it is located.

3.4 SDP Secure Workflow

In this section we will break down SDP's workflow, illustrating how all of the architecture components discussed in the previous section work together.

The following is the most basic SDP workflow for allowing an IH and AH to communicate securely:

1. The AH is cloaked by an SDP gateway on the AH or a similar construct.
2. An SDP controller is added and activated within the SDP and connected to authentication and authorization services (e.g., IAM, public key infrastructure service, device attestation, geolocation, SAML, OpenID, OAuth, LDAP, Kerberos, MFA, and identity federation).
3. An AH is added and activated within the SDP by checking into the SDP controller. It connects to and authenticates with the controller in a secure manner.
4. The IH is added and activated within the SDP, then connects to the SDP controller. The SDP controller authenticates the IH and determines a list of AH the IH is authorized to communicate with.
5. An SPA packet is always sent to establish communications, leaving the application layer cloaked from all but authorized users. In order to establish access after sending an SPA packet, the IH and AH exchange a mutual handshake using TLS for control plane communications.
6. The IH sends a login message request and receives a response from the controller.
7. The controller sends the IH a list of services available (based upon services allowed).

8. The controller also sends a message stating that the IH has been authenticated with the AH.
9. Another SPA packet is sent from the IH to the AH for data plane communications.
10. Finally, a separate mTLS handshake establishes communication for data transfers.

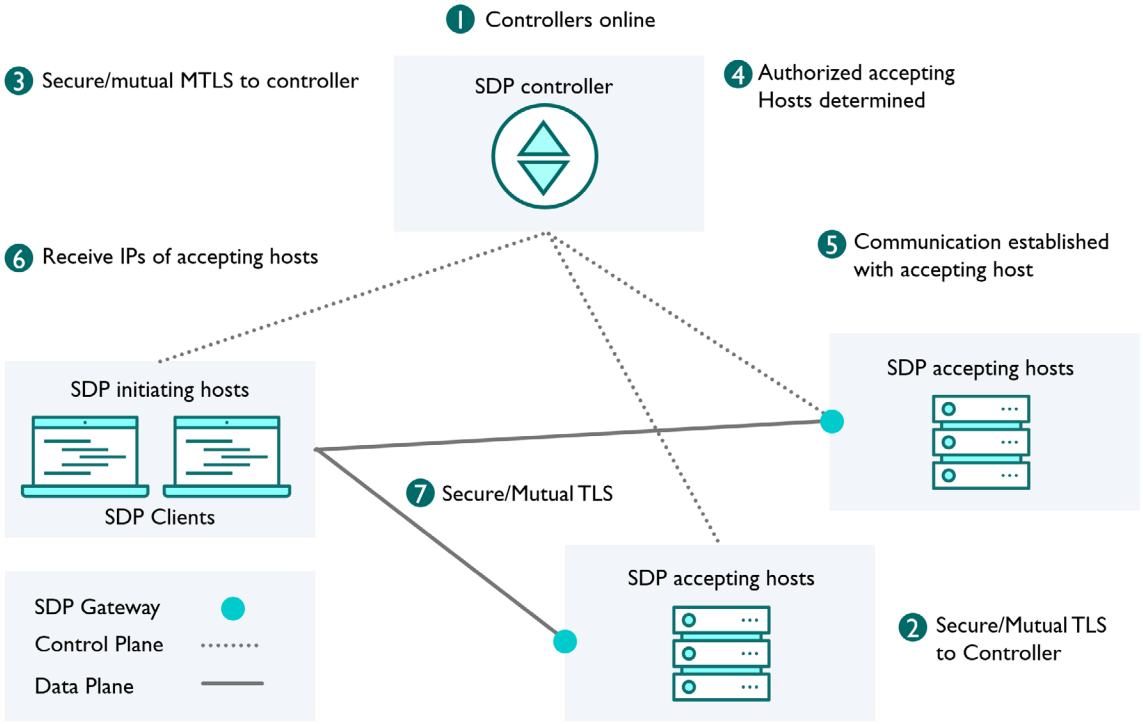

Figure 8: *SDP Secure Workflow*[15]

4 The Basics of SDP Deployment Models

This unit will cover the various SDP architectural considerations to take into account before implementation. Along with this, you will learn the basics of SDP deployment models.

4.1 Architectural Considerations

Several architectural considerations must be taken into account when deploying SDP. For example, organizations should evaluate how an SDP deployment fits into existing network topologies and technologies. Other critical considerations include how SDP impacts users, monitoring, logging, onboarding, application release, and device validation.

[15] Figure adapted from Cloud Security Allaince, "Software-Defined Perimeter (SDP) Specification v2," 10th, March, 2022, https://cloudsecurityalliance.org/artifacts/software-defined-perimeter-zero-trust-specification-v2/

4.1.1 Existing Network Topologies & Technologies

Network architects should select the SDP deployment model best suited for their particular use case. However, some models require additional in-line network components like gateways, resulting in network changes like adding firewalls or making routing alterations. This ensures that protected resources are hidden and only accessible through the SDP gateway. To fully leverage the capabilities of SDP, architects should consider proper micro-segmentation, keeping in mind that SDP ensures secure connections irrespective of the underlying network infrastructure.

Enterprise security architectures[16] can be complex, with numerous stakeholders across the organization and business units, as well as governance, risk management, and compliance (GRC) requirements alongside daily IT infrastructure operations and management. Architects should keep these factors in mind when planning their enterprise's SDP deployment.

4.1.2 Monitoring & Logging Systems

SDP affects monitoring and logging architectures. Because it uses mTLS between the IH and AH, SDP also hides network traffic from intermediary services, which may be in place to monitor for security, performance, or reliability. Architects must understand what systems are in operation and how the changes to the network traffic may affect them. However, SDP typically provides richer, identity-centric logging of user access — ideal for augmenting and enhancing existing monitoring systems for a more focused traffic monitoring scope and purpose. In addition, all dropped packets from SDP gateways and controllers can be logged, monitored, and analyzed using security tools like intrusion detection systems/intrusion detection and prevention systems (IDS/IDPS) and SIEMs. With an SDP in place, it is easier to collect the who, what, when, how, why information for every connection versus each individual packet.

4.1.3 Application Release & DevOps

High-velocity application release practices like DevOps[17] and its supporting automation and CI/CD framework require thoughtful integration with SDP. An SDP can be integrated with DevOps to secure authorized users' connections to the various deployment environments (e.g., development, test, staging, and production), as well as used during operations to ensure legitimate users have proper connectivity to protected servers and applications. Ideally, the SDP will be integrated into the application stack to fully leverage its security features. Common DevOps practices such as the use of virtualized environments and containers can further streamline SDP integration; that said, security architects must fully understand the chosen SDP deployment model and how their organization's DevOps mechanisms will interact and integrate with it. When it comes to DevOps toolset integration, security teams should carefully review and evaluate third party APIs supported by their SDP implementation.

[16] Sometimes referred to as enterprise information security architecture

[17] Cloud Security Alliance, "Enterprise Architecture Reference Guide," 18th, May 2021, https://cloudsecurityalliance.org/artifacts/enterprise-architecture-reference-guide-v2/

4.1.4 User Experience

Security teams typically strive to have their solutions work as transparently as possible, with minimal user interruption. SDP is similar to any security control where proper application of least privilege principles balances the user experience with security. Depending on the SDP deployment model, users will need to run the SDP client software on their devices. Security architects should collaborate with IT to model and plan for the user experience, client software distribution, and device onboarding processes.

4.1.5 Onboarding

The onboarding process of SDP controllers, IH, AH, and users will vary depending on the chosen deployment models. SDP systems can be managed via an API or administrative user interface.

A typical SDP onboarding process flow would involve the following steps:

1. One or more SDP controllers are brought online and connected to the appropriate optional authentication and authorization services.
2. One or more AHs are enlisted as SDP gateways. These gateways connect to and authenticate with the controllers.
3. One or more clients on the IHs are onboarded, with each user/entity authenticated by the SDP controller.

Note: Because the onboarding process is distinct from the user authentication process, users are only onboarded once but will require authentication/authorization for each subsequent connection.

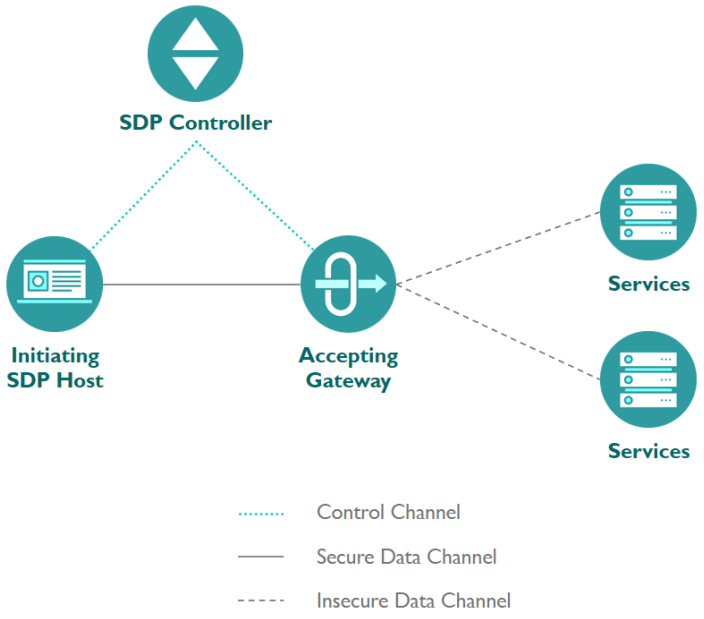

Figure 9: *Onboarding Process Flow*[18]

[18] Figure adapted from Cloud Security Allaince, "Software-Defined Perimeter (SDP) Specification v2," 10th, March, 2022, https://cloudsecurityalliance.org/artifacts/software-defined-perimeter-zero-trust-specification-v2/

4.1.6 Device Validation

mTLS proves that the device requesting access to the SDP possesses a valid, non-expired/non-revoked private key. However, this method can be compromised, as an attacker with a stolen key cannot be distinguished from a legitimate user/key. Device validation can help to further establish a trusted connection based on certificate-based keys. Per SDP, the controller acts as the trusted device because it resides in the most heavily controlled environment. The initiating and AHs must then validate themselves with the controller, thereby preventing unauthorized access via stolen keys.

4.2 Deployment Models

In this section we'll introduce the various SDP deployment models and explore their similarities and differences.

As an architecture, SDP provides the protocol to secure connections at all layers of the network stack. By deploying gateways and controllers at key locations, SDP implementers can focus on securing and protecting the most critical connections from both network-based and cross-domain attacks. All the SDP models support identity-driven network access control/authorization, and most can accommodate existing network security tools like IDS/IDPS and SIEMs by enabling the analysis of dropped packets and unsecured connections. SDP secures the connections between components, as depicted in each of the deployment models described below.

More information on these deployment models can be found in the *SDP Architecture Guide v2*[19].

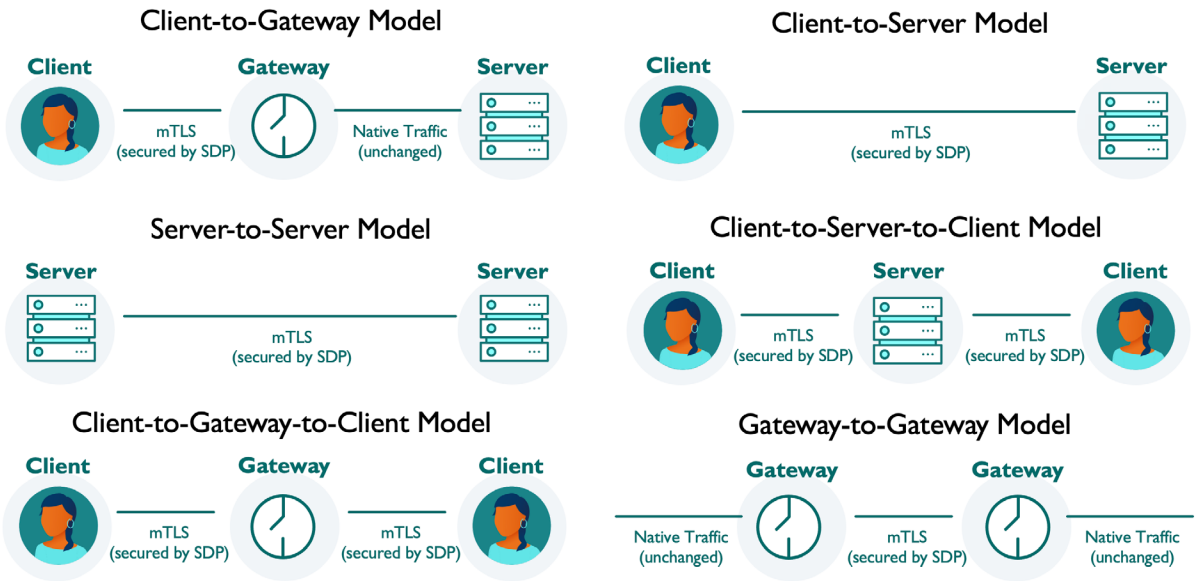

Figure 10: *SDP Deployment Models*[20]

[19] Cloud Security Alliance, "SDP Architecture Guide v2," 7th, May 2019, https://cloudsecurityalliance.org/artifacts/sdp-architecture-guide-v2/

[20] Figure adapted from Cloud Security Alliance, "SDP Architecture Guide v2," 7th, May 2019, https://cloudsecurityalliance.org/artifacts/sdp-architecture-guide-v2/

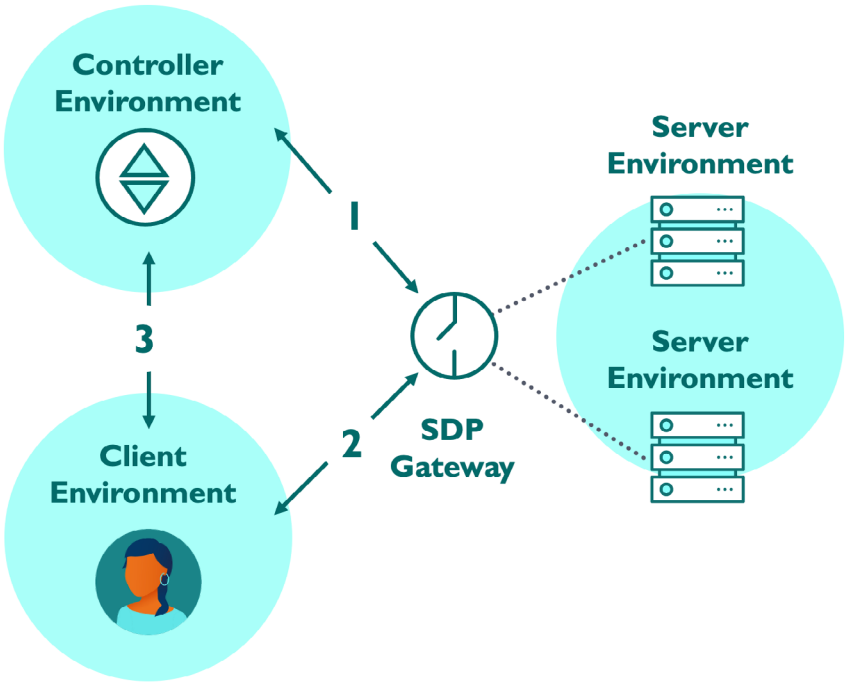

Figure 11: *Client-to-Gateway Model* [21]

4.2.1 Client-to-Gateway Model

The client-to-gateway model is suitable for use cases where one or more servers need to be protected behind a gateway. This approach is preferred when an organization is moving its applications to the cloud or securing on-premises legacy applications. The client (i.e., the IH) and gateway may be in the same location or distributed across the globe. In either case, the connections between the client and the gateway are secured, regardless of the underlying network topology.

In this model, the client is connected to the gateway directly via an mTLS tunnel where the connection terminates. To secure the connection to server environments, additional precautions must be taken. For example, the network on which the server environments reside, will need to be configured to permit inbound connections to protected servers from the gateway only. This prevents unauthorized clients from bypassing the gateway. The gateway should be configured to deny all traffic by default, and explicitly allow approved traffic. The same gateway can be used for the controller and servers by locating the controller in the cloud or near the protected servers.

This model preserves the ability for an organization to use its existing network security components, such as IDSs or IPSs, by deploying them between the SDP gateway and the protected servers.

[21] Figure adapted from Cloud Security Alliance, "SDP Architecture Guide v2," 7th, May 2019, https://cloudsecurityalliance.org/artifacts/sdp-architecture-guide-v2/

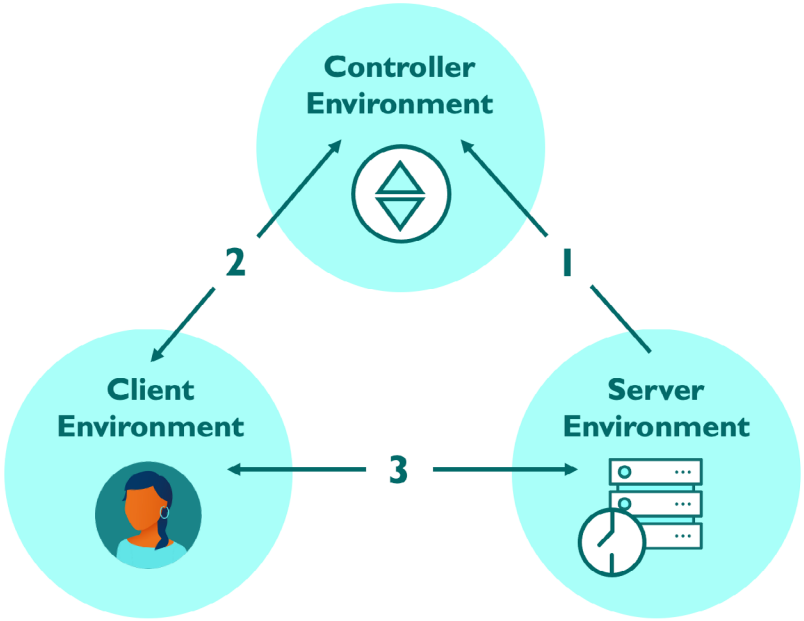

Figure 12: *Client-to-Server Model*[22]

4.2.2 Client-to-Server Model

The client-to-server model is ideal when moving applications to an IaaS provider, as it combines the server and gateway in a single host to ensure connections are secured end-to-end. Organizations are afforded a great deal of flexibility due to the portability of server-gateway combinations between multiple IaaS providers.

Client-to-server is also appropriate for securing on-premises legacy applications that cannot be upgraded. With this model, the protected servers will need to be outfitted with the gateways. The network on which the servers reside do not need configuration to restrict inbound connections to the protected servers, as the gateways or server enforcement points use SPA to prevent unauthorized connections. Secure connections to the servers provided by the gateway may be controlled by the infrastructure owner, as they have full control over the connections. Similar to the client-to-gateway model, the client may be located in the same location or distributed across the globe — in either case, it remains secured. Additionally, this model leaves the data plane completely secure, as there are no breaks in the mTLS tunnel. Traffic can be monitored by analyzing dropped packets from the SDP gateway/protected servers, thereby preserving the mTLS connections between the client and the servers.

[22] Figure adapted from Cloud Security Alliance, "SDP Architecture Guide v2," 7th, May 2019, https://cloudsecurityalliance.org/artifacts/sdp-architecture-guide-v2/

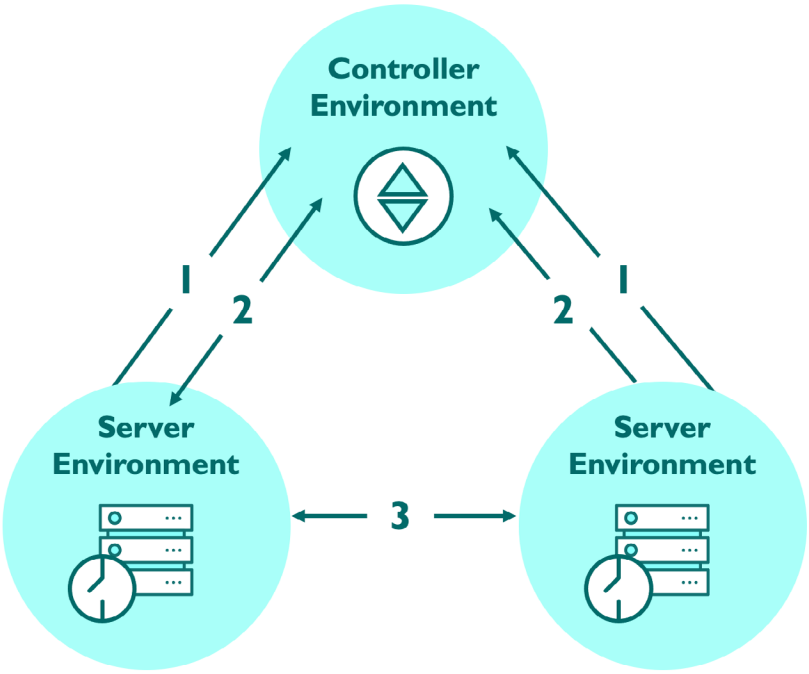

Figure 13: *Server-to-Server Model*[23]

4.2.3 Server-to-Server Model

The server-to-server model is ideal for IoT and virtual machine environments, as it offers full control over connections, regardless of where the server is located, whether in the cloud or on-premises. This model ensures that all connections between servers are encrypted, regardless of the underlying network or IP infrastructure. In addition, it ensures that all communications are explicitly permitted by an SDP allowlist policy. This model enables secure communications between servers across untrusted networks while hiding the servers from all unauthorized connections using the lightweight SPA protocol.

The server-to-server model is similar to the client-to-server model, except that the IH is itself a server and can also act as an AH. Like the client-to-server model, the server-to-server model requires that the SDP gateway, or similar lightweight technology, be installed on each server. This renders all server-to-server traffic hidden to other elements of the security ecosystem. The traffic can also be monitored by analyzing all the dropped packets from the SDP gateway/protected servers. The secure connections to the servers going through the gateway are under the control of the owner of the application/services on the server by default, giving the owner full control of these connections.

[23] Figure adapted from Cloud Security Alliance, "SDP Architecture Guide v2," 7th, May 2019, https://cloudsecurityalliance.org/artifacts/sdp-architecture-guide-v2/

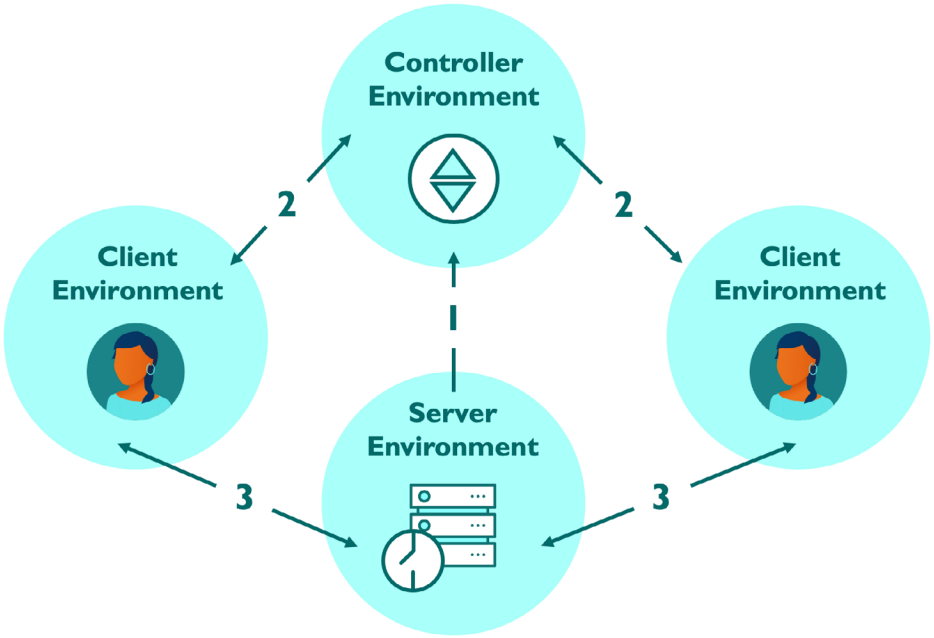

Figure 14: *Client-to-Server-to-Client Model*[24]

4.2.4 Client-to-Server-to-Client Model

The client-to-server-to-client model is well-suited for environments in which organizations are moving their peer-to-peer applications to the cloud, such as IP telephone, chat, or videoconferencing. Regardless of where the server environment is located (cloud or on-premises), organizations can have full control over the connections to the clients. This model results in a logical peer-to-peer relationship between two clients. This can be used for applications in which the traffic must pass through an intermediary server. In these cases, the SDP conceals the IP addresses of the connecting clients, encrypts the network connections between the components, and protects the server/AH from unauthorized network connections by using SPA.

[24] Figure adapted from Cloud Security Alliance, "SDP Architecture Guide v2," 7th, May 2019, https://cloudsecurityalliance.org/artifacts/sdp-architecture-guide-v2/

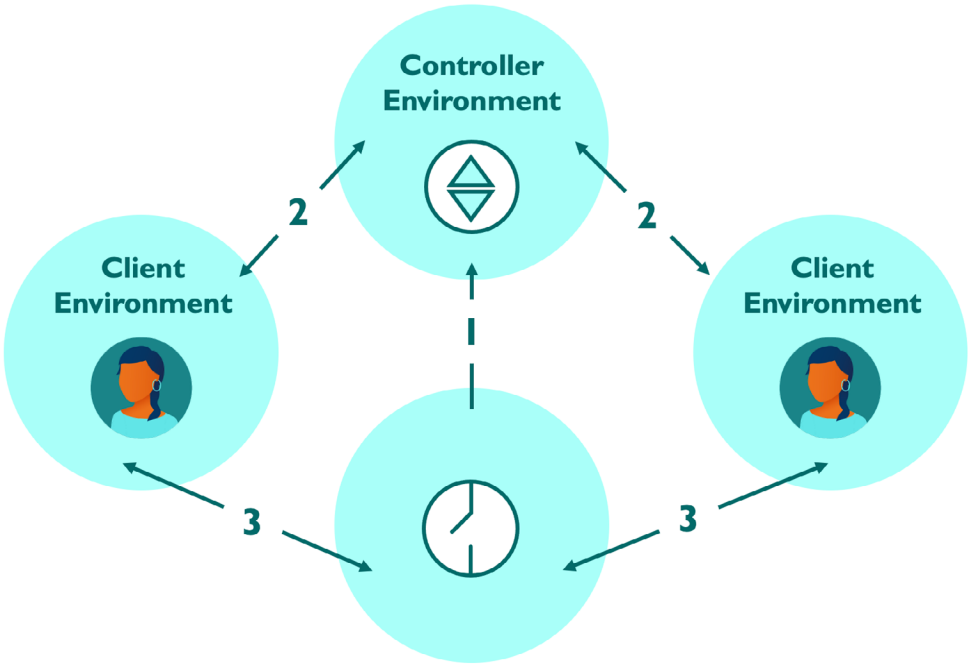

Figure 15: *Client-to-Gateway-to-Client Model*[25]

4.2.5 Client-to-Gateway-to-Client Model

This variation of the client-to-server-to-client model has the advantage of supporting peer-to-peer network protocols that require clients to connect directly to one another, while still enforcing SDP access policies. This results in a logical connection between the clients, each acting as either IH, AH, or both depending on the application protocol. It's worth noting that while the application protocol determines how the clients connect to each other, the SDP gateway continues to perform its standard role as a firewall.

[25] Figure adapted from Cloud Security Alliance, "SDP Architecture Guide v2," 7th, May 2019, https://cloudsecurityalliance.org/artifacts/sdp-architecture-guide-v2/

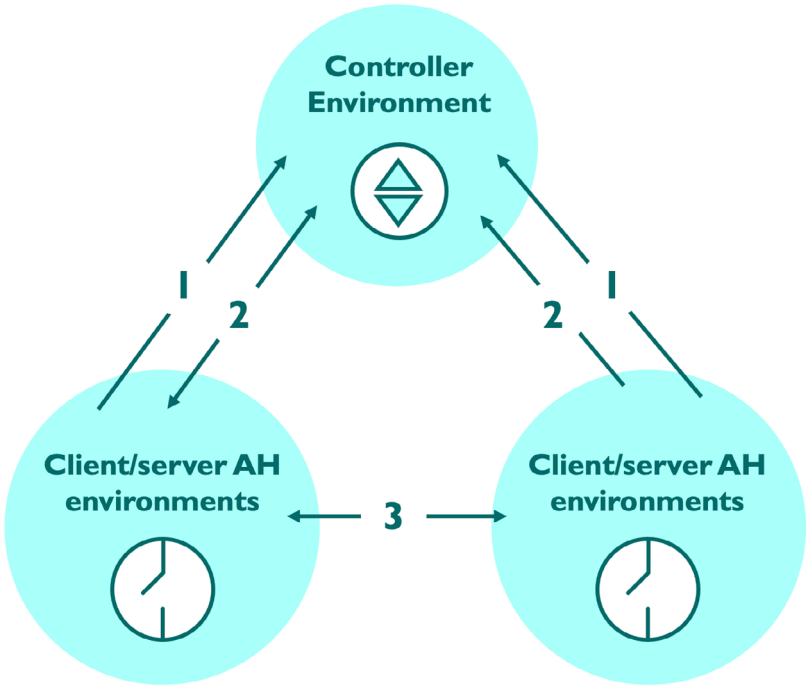

Figure 16: *Gateway-to-Gateway Model*[26]

4.2.6 Gateway-to-Gateway Model

The gateway-to-gateway model is well-suited for certain IoT environments. In this scenario, one or more servers sits behind the AH and acts as a gateway between the clients and the servers. At the same time, one or more clients sits behind an IH that acts as a gateway.

In this SDP model, the IH gateway is running SDP client software, but the client devices are not — they may be incapable of supporting SDP client installation, such as in the case of printers, scanners, sensors, or IoT devices. In this model, the gateway would operate as a firewall or router/proxy, depending on the implementation.

[26] Figure adapted from Cloud Security Alliance, "SDP Architecture Guide v2," 7th, May 2019, https://cloudsecurityalliance.org/artifacts/sdp-architecture-guide-v2/

Conclusion

In this introductory SDP course, we provided learners with an overview of SDP's history and how it relates to ZT. We defined key SDP terminology and principles, explored its myriad of technology and business benefits, and walked through current security architecture issues that SDP addresses. Learners were introduced to leading industry cyber risk matrices/lists in order to illustrate how SDP addresses specific, common threats, followed by a deeper dive into its core tenets and underlying technologies.

Lastly, learners were provided with a set of crucial architectural considerations to account for when implementing SDP, followed by the various SDP deployment options and related guidance for selecting the appropriate model.

Glossary

For additional terms, please refer to our Cloud Security Glossary, a comprehensive glossary that combines all the glossaries created by CSA Working Groups and research contributors into one place.

Term	Definition	Source
802.1x	An IEEE standard for local and metropolitan area networks–Port-Based Network Access Control. IEEE 802 LANs are deployed in networks that convey or provide access to critical data, that support mission critical applications, or that charge for service. Port-based network access control regulates access to the network, guarding against transmission and reception by unidentified or unauthorized parties, and consequent network disruption, theft of service, or data loss.	https://1.ieee802.org/security/802-1x/
Accepting Host (AH)	The SDP policy enforcement points (PEPs) that control access to any resource (or service) to which an identity might need to connect, and to which the responsible enterprise needs to hide and control access. AHs can be located on-premises, in a private cloud, public cloud, etc.	https://cloudsecurityalliance.org/artifacts/software-defined-perimeter-zero-trust-specification-v2/
Access	To make contact with one or more discrete functions of an online, digital service.	https://csrc.nist.gov/glossary/term/access
Active Directory (AD)	A Microsoft directory service for the management of identities in Windows domain networks.	https://csrc.nist.gov/glossary/term/active_directory
Application Programming Interface (API)	A system access point or library function that has a well-defined syntax and is accessible from application programs or user code to provide well-defined functionality.	https://csrc.nist.gov/glossary/term/application_programming_interface

Attribute-Based Access Control (ABAC)	An access control approach in which access is mediated based on attributes associated with subjects (requesters) and the objects to be accessed. Each object and subject has a set of associated attributes, such as location, time of creation, access rights, etc. Access to an object is authorized or denied depending upon whether the required (e.g., policy-defined) correlation can be made between the attributes of that object and of the requesting subject.	https://csrc.nist.gov/glossary/term/abac
Authentication	Verifying the identity of a user, process, or device, often as a prerequisite to allowing access to resources in an information system.	https://csrc.nist.gov/glossary/term/authentication
Authorization	The right or a permission that is granted to a system entity to access a system resource.	https://csrc.nist.gov/glossary/term/authorization
Brute Force Attacks	An attempt to discover a password by systematically trying every possible combination of letters, numbers, and symbols until you discover the one correct combination that works.	https://owasp.org/www-community/controls/Blocking_Brute_Force_Attacks
Certificate Forgery	Data transmitted from an online certificate issuing server to output devices (such as a PC or printer) can be accessed by a hacker and modified into a false certificate.	https://ieeexplore.ieee.org/document/6922060
Client-to-Authenticator Protocol (CTAP)	An application layer protocol for communication between a roaming authenticator and another client/platform, as well as bindings of this application protocol to a variety of transport protocols using different physical media. The application layer protocol defines requirements for such transport protocols.	https://fidoalliance.org/specs/fido-v2.0-id-20180227/fido-client-to-authenticator-protocol-v2.0-id-20180227.html
Control Plane	Used by various infrastructure components (both enterprise-owned and from service providers) to maintain and configure assets; judge, grant, or deny access to resources; and perform any necessary operations to set up communication paths between resources.	https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207.pdf

Controller (SDP Controller)	Determines which SDP hosts can communicate with each other. The controller may relay information to external authentication services such as attestation, geo-location, and/or identity servers.	https://downloads.cloudsecurityalliance.org/initiatives/sdp/Software_Defined_Perimeter.pdf
Data Plane	Used for communication between software components. This communication channel may not be possible before the path has been established via the control plane.	https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-207.pdf
Device Attestation	The ability to provide proof that elements of the device (e.g., firmware) have not been tampered with.	https://nvlpubs.nist.gov/nistpubs/CSWP/NIST.CSWP.09082020-draft.pdf
Device Onboarding Process	Involves the installation of the physical device and the setup of credentials so that it can securely communicate with its target cloud or platform. T	https://media.fidoalliance.org/wp-content/uploads/2021/04/Introduction-to-FIDO-Device-Onboard-1.pdf
Distributed Denial-of-Service (DDoS)	Involves multiple computing devices in disparate locations sending repeated requests to a server with the intent to overload it and ultimately render it inaccessible.	https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.1800-15.pdf
Domain Name System (DNS) Poisoning	Results in a DNS resolver storing (i.e., caching) invalid or malicious mappings between symbolic names and IP addresses.	https://www.cs.cornell.edu/~shmat/shmat_securecomm10.pdf
Firewall	An inter-network connection device that restricts data communication traffic between two connected networks. A firewall may be either an application installed on a general-purpose computer or a dedicated platform (appliance), which forwards or rejects/drops packets on a network. Typically firewalls are used to define zone borders. Firewalls generally have rules restricting which ports are open.	https://csrc.nist.gov/glossary/term/firewall
Gateway (SDP Gateway)	Provides authorized users and devices with access to protected processes and services. The gateway can also enact monitoring, logging, and reporting on these connections.	https://cloudsecurityalliance.org/artifacts/sdp-architecture-guide-v2/
Geolocation	Provides access to geographical location information associated with the hosting device.	https://www.w3.org/TR/geolocation/
Hash Message Authentication Code (HMAC)	A message authentication code that uses a cryptographic key in conjunction with a hash function.	https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.198-1.pdf

Hypertext Transport Protocol Secure (HTTPS)	A secure network communication method, technically not a protocol in itself, HTTPS is the result of layering the Hypertext Transfer Protocol (HTTP) on top of the SSL/TLS protocol, thus adding the security capabilities of SSL/TLS to standard HTTP communications.	https://iapp.org/resources/article/hypertext-transfer-protocol-secure/
Identity (ID)	The set of attribute values (i.e., characteristics) by which an entity is recognizable and that, within the scope of an identity manager's responsibility, is sufficient to distinguish that entity from any other entity.	https://csrc.nist.gov/glossary/term/identity
Identity and Access Management (IAM)	The set of technology, policies, and processes that are used to manage access to resources.	https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-203.pdf
Identity Provider (IdP)	A trusted entity that issues or registers subscriber authenticators and issues electronic credentials to subscribers. A cloud service provider may be an independent third party or issue credentials for its own use.	https://csrc.nist.gov/glossary/term/identity_provider
Initiating Host (IH)	The host that initiates communication to the controller and to the AHs.	https://downloads.cloudsecurityalliance.org/initiatives/sdp/SDP_Specification_1.0.pdf
Keyloggers	A reconnaissance tool--with keylogging and screen capture functionality--used for information gathering on compromised systems.	https://attack.mitre.org/software/
Lightweight Directory Access Protocols (LDAP)	A networking protocol for querying and modifying directory services running over TCP/IP.	https://csguide.cs.princeton.edu/email/setup/ldap
Man-in-the-middle (MITM) attacks	An attack where the adversary positions himself in between the user and the system so that he can intercept and alter data traveling between them.	https://csrc.nist.gov/glossary/term/mitm

Micro-segmentation	Is the technique of creating secure zones within a data center and cloud deployments that allow the organization to separate and secure each workload. This makes network security more granular and effective. These secure zones are created based on business services, and rules are defined to secure information workflow.	https://www.techtarget.com/searchnetworking/definition/microsegmentation
Misconfiguration	An incorrect or suboptimal configuration of an information system or system component that may lead to vulnerabilities.	https://csrc.nist.gov/glossary/term/misconfiguration
Multi-factor Authentication (MFA)	Authentication using two or more factors to achieve authentication. Factors include: (i) something you know (e.g., password/ personal identification number (PIN)); (ii) something you have (e.g., cryptographic identification device, token); or (iii) something you are (e.g., biometric).	https://csrc.nist.gov/glossary/term/multi_factor_authentication
Multiprotocol Label Switching (MPLS)	An Internet Engineering Task Force (IETF)-specified framework that provides for the efficient designation, routing, forwarding, and switching of traffic flows through the network. MPLS performs the following functions: specifies mechanisms to manage traffic flows of various granularities, remains independent of the Layer-2 and Layber-3 protocols, provides a means to map IP addresses to simple, fixed-length labels used by different packet-forwarding and packet-switching technologies, interfaces to existing routing protocols, and supports the IP, ATM, and frame-relay Layer-2 protocols.	http://tele1.dee.fct.unl.pt/rit1_2020_2021/pages/IEC_MPLS.pdf
Mutual Transport Layer Security (mTLS)	An approach where each microservice can identify who it talks to, in addition to achieving confidentiality and integrity of the transmitted data. Each microservice in the deployment has to carry a public/private key pair and uses that key pair to authenticate to the recipient microservices via mTLS.	https://cheatsheetseries.owasp.org/cheatsheets/Microservices_security.html#mutual-transport-layer-security
Network Access Control (NAC)	A method of bolstering the security of a private or "on-premise" network by restricting the availability of network resources to endpoint devices that comply with a defined security policy.	https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-41r1.pdf

Network Address Translation (NAT)	A function by which internet protocol addresses within a packet are replaced with different IP addresses. This function is most commonly performed by either routers or firewalls. It enables private IP networks that use unregistered IP addresses to connect to the internet. NAT operates on a router, usually connecting two networks together, and translates the private (not globally unique) addresses in the internal network into legal addresses before packets are forwarded to another network.	https://csrc.nist.gov/glossary/term/network_address_translation
Network Segmentation	Splitting a network into sub-networks, for example, by creating separate areas on the network which are protected by firewalls configured to reject unnecessary traffic. Network segmentation minimizes the harm of malware and other threats by isolating it to a limited part of the network.	https://www.nist.gov/itl/smallbusinesscyber/cybersecurity-basics/glossary
Next Generation Firewall (NGFW)	Deep-packet inspection firewalls that move beyond port/protocol inspection and blocking to add application-level inspection, intrusion prevention, and bringing intelligence from outside the firewall. An NGFW should not be confused with a stand-alone network intrusion prevention system (IPS), which includes a commodity or non enterprise firewall, or a firewall and IPS in the same appliance that are not closely integrated.	https://www.gartner.com/en/information-technology/glossary/next-generation-firewalls-ngfws
Open Systems Interconnection (OSI)	Qualifies standards for the exchange of information among systems that are "open" to one another for this purpose by virtue of their mutual use of applicable standards.	https://www.ecma-international.org/wp-content/uploads/s020269e.pdf

Pass-The-Hash	Adversaries may "pass the hash" using stolen password hashes to move laterally within an environment, bypassing normal system access controls. Pass the hash (PtH) is a method of authenticating as a user without having access to the user's cleartext password. This method bypasses standard authentication steps that require a cleartext password, moving directly into the portion of the authentication that uses the password hash.	https://attack.mitre.org/techniques/T1550/002/
Pass-The-Ticket	Adversaries may "pass the ticket" using stolen Kerberos tickets to move laterally within an environment, bypassing normal system access controls. Pass the ticket (PtT) is a method of authenticating to a system using Kerberos tickets without having access to an account's password. Kerberos authentication can be used as the first step to lateral movement to a remote system.	https://attack.mitre.org/techniques/T1550/003/
Phishing	A technique for attempting to acquire sensitive data, such as bank account numbers, through a fraudulent solicitation in email or on a web site, in which the perpetrator masquerades as a legitimate business or reputable person.	https://csrc.nist.gov/glossary/term/phishing
Port	Another essential asset through which security can be breached. In computer science, ports are of two types - physical ports (which is a physical docking point where other devices connect) and logical ports (which is a well-programmed docking point through which data flows over the internet). Security and its consequences lie in a logical port.	https://www.w3schools.in/cyber-security/ports-and-its-security/

Public Key Infrastructure (PKI)	The framework and services that provide for the generation, production, distribution, control, accounting, and destruction of public key certificates. Components include the personnel, policies, processes, server platforms, software, and workstations used for the purpose of administering certificates and public-private key pairs, including the ability to issue, maintain, recover, and revoke public key certificates.	https://csrc.nist.gov/glossary/term/public_key_infrastructure
Role Based Access Control (RBAC)	Access control based on user roles (i.e., a collection of access authorizations a user receives based on an explicit or implicit assumption of a given role). Role permissions may be inherited through a role hierarchy and typically reflect the permissions needed to perform defined functions within an organization. A given role may apply to a single individual or to several individuals.	https://csrc.nist.gov/glossary/term/role_based_access_control
Security Assertion Markup Language (SAML)	A protocol consisting of XML-based request and response message formats for exchanging security information, expressed in the form of assertions about subjects, between online business partners.	https://csrc.nist.gov/glossary/term/security_assertion_markup_language
Security Group	Are sets of IP filter rules that are applied to all project instances, which define networking access to the instance.	https://docs.openstack.org/nova/train/admin/security-groups.html#:~:text=Security%20groups%20are%20sets%20of,networking%20access%20to%20the%20instance.&text=By%20default%2C%20security%20groups%20(and,by%20the%20Neutron%20networking%20service
Single Packet Authorization (SPA)	Can authenticate a user to a system for simple remote administration. It is a protocol for allowing a remote user to authenticate securely on a "closed" system (limited or no open services) and make changes to or run applications on the "closed" system.	https://www.blackhat.com/presentations/bh-usa-05/bh-us-05-madhat.pdf

Software-Defined Network (SDN)	An approach to computer networking that allows network administrators to manage network services through abstractions of higher-level functionality. SDNs manage the networking infrastructure. This is done by decoupling the system that makes decisions about where traffic is sent (the control plane) from the underlying systems that forward traffic to the selected destination (the data plane).	https://ieeexplore.ieee.org/abstract/document/6819788
Software-Defined Perimeter (SDP)	A network security architecture that is implemented to provide security at Layers 1-7 of the OSI network stack. An SDP implementation hides assets and uses a single packet to establish trust via a separate control and data plane prior to allowing connections to hidden assets.	https://cloudsecurityalliance.org/artifacts/software-defined-perimeter-and-zero-trust/
Structured Query Language (SQL) Injection	These attacks, which are still quite common on the Internet, look for web sites that pass insufficiently processed user input to database back-ends and then send carefully-crafted input that will cause exposure of database records, and possibly allow destruction of databases.	https://nvlpubs.nist.gov/nistpubs/Legacy/IR/nistir7682.pdf
Transmission Control Protocol (TCP)	A transport protocol that is used on top of IP to ensure reliable transmission of packets. TCP includes mechanisms to solve many of the problems that arise from packet-based messaging, such as lost packets, out of order packets, duplicate packets, and corrupted packets. Since TCP is the protocol used most commonly on top of IP, the Internet protocol stack is sometimes referred to as TCP/IP.	https://rb.gy/qcorbs
Transmission Control Protocol/ Internet Protocol (TCP/IP)	A set of protocols covering (approximately) the network and transport layers of the seven-layer Open Systems Interconnection (OSI) network model.	https://www.gartner.com/en/information-technology/glossary/tcpip-transmission-control-protocolinternet-protocol
Transport Layer Security (TLS)	A cryptographic protocol, successor to SSL, that provides security for communications over a computer or IP network.	https://csrc.nist.gov/glossary/term/transport_layer_security

Virtual Local Area Network (VLAN)	A broadcast domain that is partitioned and isolated within a network at the data link layer. A single physical local area network (LAN) can be logically partitioned into multiple, independent VLANs; a group of devices on one or more physical LANs can be configured to communicate within the same VLAN, as if they were attached to the same physical LAN.	https://csrc.nist.gov/glossary/term/virtual_local_area_network_vlan
Virtual Private Network (VPN)	A virtual network built on top of existing physical networks that can provide a secure communications mechanism for data and IP information transmitted between networks or between different nodes on the same network.	https://csrc.nist.gov/glossary/term/virtual_private_network
Web Authentication (WebAuth)	Web Authentication (WebAuthn), a core component of FIDO Alliance's FIDO2 set of specifications, is a web-based API that allows websites to update their login pages to add FIDO-based authentication on supported browsers and platforms. FIDO2 enables users to leverage common devices to easily authenticate to online services in both mobile and desktop environments.	https://fidoalliance.org/fido2/fido2-web-authentication-webauthn/