Integrating SDP and DNS

Enhanced Zero Trust Policy Enforcement

cloud security alliance®

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

Acknowledgments

Lead Authors

Jason Garbis
Juanita Koilpillai
Srikrupa Srivatsan
PG Menon

Contributors

Michael Roza

Reviewers

Nader Zaveri
Andrea Knoblauch

CSA Global Staff

Shamun Mahmud

Dedication

This paper is dedicated to the memory of Juanita Koilpillai, a security leader, influencer, mentor, and friend.

The Software-Defined Perimeter (SDP) and Zero Trust Working Group is a Cloud Security Alliance (CSA) research working group that promotes adopting zero trust security principles, providing practical and technically sound guidance on how organizations can and should approach these principles for their cloud and non-cloud environments. This group will build on and leverage the National Institute of Standards and Technology (NIST) Zero Trust research and approach. The group will also promote SDP as a recommended architecture for achieving zero trust benefits and principles. In addition, it will revise and expand the SDP specification to capture and codify the knowledge gained from past experiences. Finally, while promoting and recommending SDP, the group will take an inclusive approach to alternative security architectures and objectively support them if they are aligned with the zero trust philosophy.

Table of Contents

1. Introduction 5
    1.1 Purpose 5
    1.2 Scope 5
    1.3 Audience 6
    1.4 The Domain Name System (DNS) 6
        1.4.1 Dynamic Host Configuration Protocol (DHCP) 7
        1.4.2 Internet Protocol Address Management (IPAM) 8
        1.4.3 Cloud Managed Implementation 8
    1.5 DNS-Based Security 9
        1.5.1 Malware Control Point 9
        1.5.2 Blocking Data Exfiltration 10
        1.5.3 Domain Generation Algorithms (DGAs) Control Point 11
        1.5.4 Category-Based Filtering 12
    1.6 Zero Trust Policy Enforcement 13
        1.6.1 SDP and Zero Trust Policy Enforcement 14
        1.6.2 DNS and Zero Trust Policy Enforcement 15
2. SDP/Zero Trust and DNS Use Cases 17
    2.1 Use Case #1: DNS providing Context and Metadata to SDP 17
        2.1.1 Policy Enforcement in Use Case No. 1 19
            2.1.1.1 Network Context and Identity Information 19
        2.1.2 Responding to Malicious Activity 21
        2.1.3 Location-Based Access Controls 22
        2.1.4 Device-Based Access Controls 22
        2.1.5 User-Based Access Control 23
    2.2 Use Case #2 - SDP Controller Publishing Policy Decisions to DNS 23
        2.2.1 Policy Enforcement in DNS - an additional layer of security 24
3. Conclusion 25
4. References 26
5. Acronyms 28

1. Introduction

Against a backdrop of soaring network complexities and heightened threat landscapes, organizations require solutions that simplify, optimize and secure network interactions. The domain name system (DNS) maps human-readable domain names (e.g., cloudsecurityalliance.org) to internet protocol (IP) addresses—functionality that is critical to reliable internet operation and connectivity. Nearly every network connection begins with a DNS query, whether from a user's web browser, a connected device, or a business server. Unfortunately, DNS' ubiquity and the largely open, connectionless, and unencrypted nature of the protocol makes DNS a commonly exploited means of infiltrating malware into networks and exfiltrating data (often unnoticed). However, organizations can integrate a software-defined perimeter (SDP) architecture with DNS. This strategy results in improved security, leveraging DNS as a zero trust network policy enforcement point alongside the SDP policy enforcement points—and mining valuable DNS data for faster threat response by SDPs.

Two other core network services that are necessary to scaling network connectivity are dynamic host configuration protocol (DHCP), and internet protocol address management (IPAM). The three core network services are collectively referred to as DDI (DNS, DHCP, IPAM). Integrating these three components helps provide control, automation, and security for today's modern and highly distributed networks. The DDI combination has the unique advantage of logging who's on the network, where they are going, and (more importantly) where they have been. When coupled with threat intelligence feeds, DDI systems have access to information to set and enforce policy at the control plane and DNS layers.

The benefit of setting and enforcing policy at the DNS layer is that it is not compute-intensive and can scale to millions. However, it is vital to note that policies at the DNS layer are coarse (e.g., domain names). Therefore, additional mechanisms are required for a fine-grained policy framework and enforcement to leverage the DDI database. DDI services can provide enterprises with visibility and control, and—when combined with the software-defined perimeter—they can deliver considerably improved security and help organizations proceed on their zero trust security journeys.

1.1 Purpose

The purpose of this research document is to explain how DNS and the enterprise-managed DDI system can be combined with a software-defined perimeter to deliver improved security visibility, resiliency, and responsiveness.

1.2 Scope

This position paper explores two use cases where enterprise-managed DDI integrates with SDP to improve security, contextual awareness, and responsiveness. This type of integration—tying together systems traditionally distinct for more holistic enforcement—is a hallmark of the zero trust security approach. This paper does not address the DNS infrastructure security itself.

1.3 Audience

The target audience for this document includes:

- Enterprise architects and security leaders deploying zero trust/SDP products who want to take a holistic approach.
- Security and IT practitioners responsible for enterprise DNS, DHCP, and IPAM systems seeking ways to improve systems' security effectiveness.
- Vendors or technology providers implementing zero trust/SDP architecture within their products or solutions.

1.4 The Domain Name System (DNS)

DNS is a hierarchical naming system built on a distributed database for computers, services, or any resource connected to the internet or a private network. DNS translates domain names into numerical identifiers associated with networking equipment to locate and address devices worldwide. Analogous to a system "phone book," DNS allows browsers to translate, for example, "https://cloudsecurityalliance.org" to the actual IP address of CSA's web servers.

Public DNS is offered to all internet users and functions as a recursive name server. Public DNS examples include Cisco OpenDNS, Google Public DNS, Cloudflare, and public DNS servers operated by most internet service providers (ISPs). However, public DNS servers or DNS included in many internet service bundles are typically not suited for businesses. Often, organizations use commercial or open-source DNS servers as private DNS servers for enhanced control, security, reliability, and internal use speed.

There are several different DNS server types described in these references.[1,2] They must work together to resolve domain names to IP addresses. If DNS servers don't possess these attributes, they may query other DNS servers (an operation called a "recursion"). The recursion process is shown in Figure 1: Recursive DNS Resolution in the Enterprise.

1 Johnson, D. (2021, February 16). *What is a DNS server? How Domain Name System servers connect you to the internet*. Business Insider. Retrieved March 9, 2022, from https://www.businessinsider.com/what-is-a-dns-server?r=US&IR=T

2 OmniSecu.com. (n.d.). *Recursive and Iterative DNS Queries*. Retrieved March 9, 2022, from https://www.omnisecu.com/tcpip/recursive-and-iterative-dns-queries.php

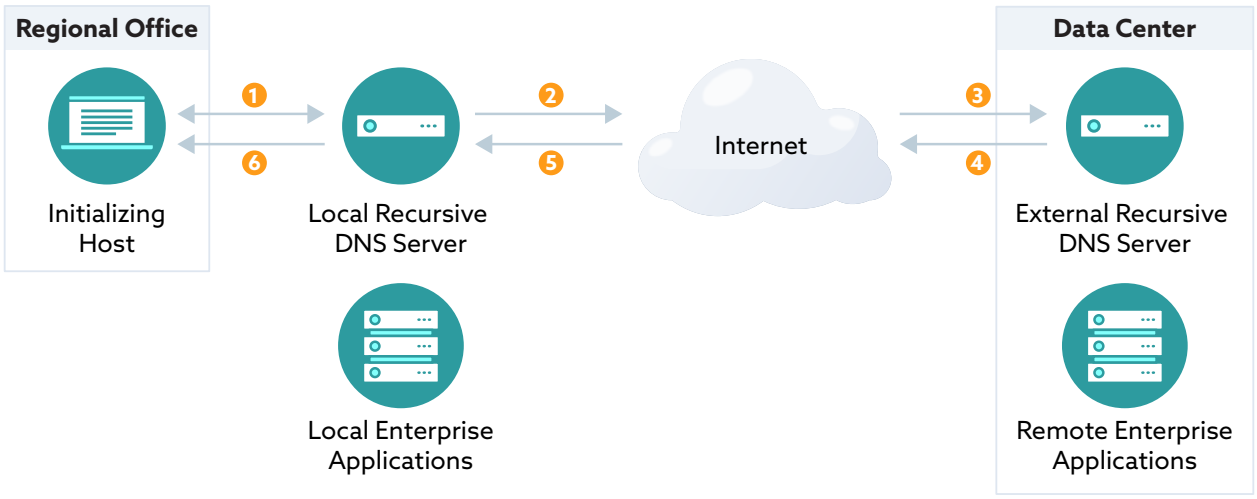

Figure 1: Recursive DNS Resolution in the Enterprise

#	Movement	Description
1	Client to Local DNS Server	Client (initiating host) sends DNS query to local DNS server to get the IP address of the application it seeks. Local DNS server responds if it has the information (i.e., for a local enterprise application).
2	Local DNS Server to Internet	If the local DNS server does NOT have the information, it forwards the query to other DNS servers.
3	Internet to External DNS Server	All domains are registered per the domain name hierarchy. The DNS servers resolve the query, potentially through additional layers of recursion.
4	External DNS Server to Internet	The DNS response is sent back over the internet.
5	Internet to Local DNS Server	The local DNS server caches this response for future use.
6	Local DNS Server to Client	The local DNS server forwards the response to the client.

1.4.1 Dynamic Host Configuration Protocol (DHCP)

DHCP is an automatic configuration protocol service that assigns IP addresses to network devices at the moment of connection and is essential for connecting a device to a network. Every device must have an IP address to communicate. DHCP allows devices to be configured automatically, eliminating the need for intervention by a network administrator, and provides a central database for keeping track of devices connected to the network. In addition, DHCP prevents two devices from accidentally being configured with the same IP address.

DHCP and DNS, operate at open systems interconnection (OSI) "layer 7 or the Application Layer".

1.4.2 Internet Protocol Address Management (IPAM)

Internet protocol address management is a system enterprises use to administer DNS and DHCP on private networks. The IPAM system is a means of planning, tracking, and managing how IP addresses are assigned and resolved within a network. Most commonly, technologies such as DNS and DHCP are used in tandem to perform these tasks. However, a well-architected DDI with IPAM integrates DNS and DHCP service data so that each service is aware of changes in the other service.

For example, DNS knows the IP address assigned to a client via DHCP and then updates itself accordingly. Additionally, knowing the IP address assigned to a client allows a private DNS to resolve clients by hostname, even when an IP address is assigned dynamically by DHCP.

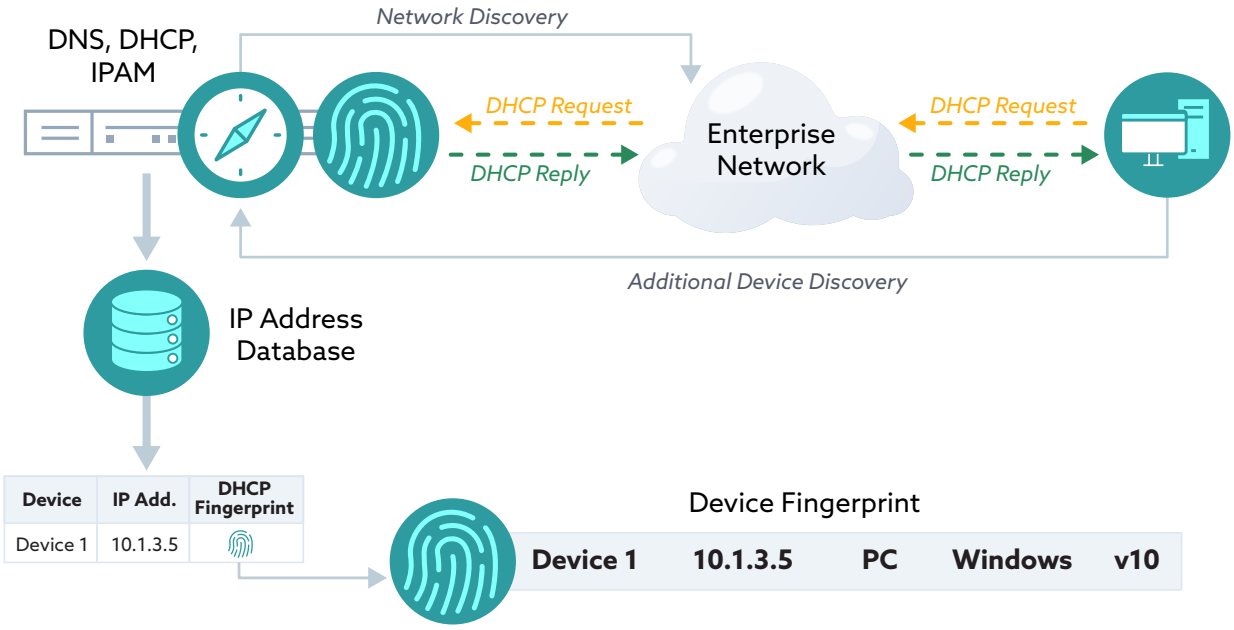

Figure 2: Contextual Information Available in DDI

1.4.3 Cloud Managed Implementation

DNS has traditionally been deployed and managed on-premises. However, like many enterprise IT and security infrastructure elements, the deployment model for the enterprise-operated DDI has shifted toward being cloud-managed.

In cloud-managed DDI, the management and control plane are moved to the cloud, and a lightweight protocol engine is placed on-premise. This lightweight protocol engine may be containerized or a virtual machine. Local presence provides local survivability. Drilling machines with hundreds of sensors at remote sites can continue operations, or manufacturing centers can continue production even if they are cut off from the internet. Moving the control and management functions to the cloud delivers several software-as-a-service (SaaS)-related benefits. These include:

- Life cycle management is automated.
- The consumption model becomes more SaaS-like.
- There is no need to over-provision.
- Organizations can scale their usage up or down based on business needs.
- And lastly, unlike on-premise management, cloud-based management scales to hundreds or thousands of remote sites such as gas stations and retail stores.

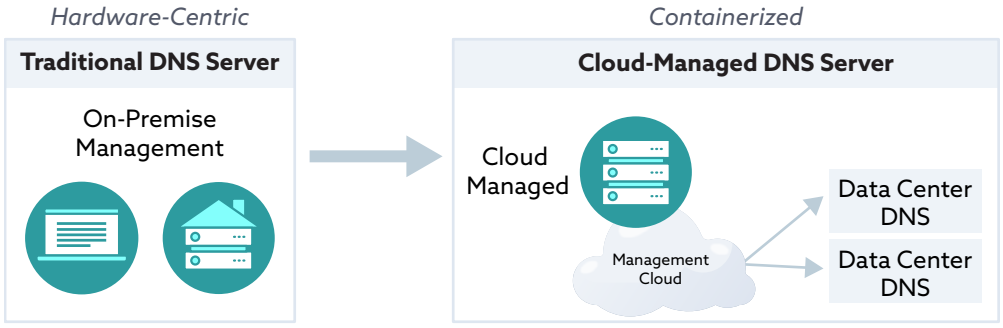

Figure 3: Cloud-Managed Private DNS

1.5 DNS-Based Security

Domain name system-based security is crucial for the early detection and blocking of malware activity within networks. Malware often needs to utilize DNS to resolve IP addresses of any command and control (CnC) servers. Domain name systems are also used by malware as a covert communications channel to avoid detection by enterprise security.

1.5.1 Malware Control Point

When namespace resolution is needed, DNS is the first point of communication from a compromised device propagating malware. This means that DNS has a front-row seat to malware activity and can detect and respond to malware attacks. By using high-quality aggregated threat intelligence on DNS servers, organizations can break the CnC channel and prevent the execution of malware campaigns, including ransomware campaigns. This is achievable by using a response policy zone (or RPZ), which effectively blocks DNS resolution to external domains (e.g., webdisk.yakimix[.]com) that are known to be malicious or are known CnC servers. Threat intel can include compromised hostnames, domain names, and URLs. This information can be used to block DNS resolution to these destinations.

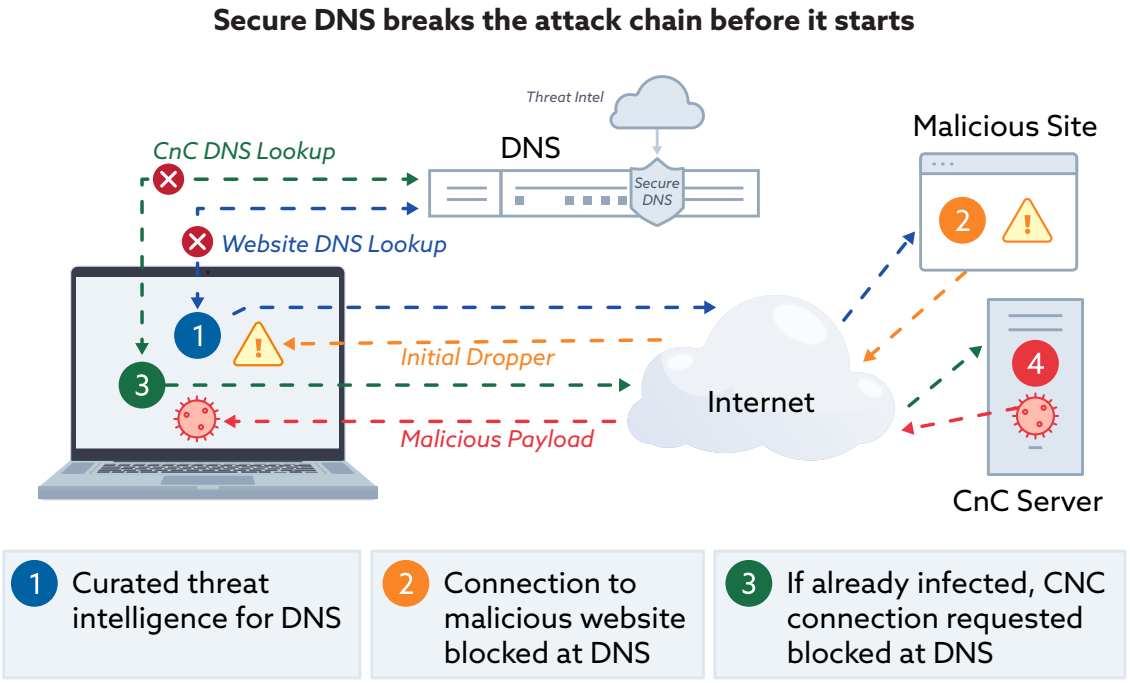

Figure 4: Using Threat Intel on DNS for Blocking Malware Activity

1.5.2 Blocking Data Exfiltration

DNS can also act as a backchannel for exfiltrating data out of an enterprise. Bad actors can use standard tunneling toolkits or custom methods to bypass traditional security technologies (e.g., firewalls, intrusion detection systems (IDSs), etc.). For example, DNS tunneling was used by the recent Ryuk ransomware campaign that targeted healthcare organizations.[3] In this scenario, clever hackers realized they could secretly communicate with a target computer by sneaking commands and data into validly formatted DNS requests. This idea is at the core of DNS Tunneling.[4,5,6] Unfortunately, variants of tunneling are hard to detect because they use previously unknown methods and could be going to destinations that don't have poor reputations yet (meaning they are not classified as malicious and won't show up in any threat feeds). The best way to mitigate zero-day data exfiltration/tunneling is to use artificial intelligence/machine learning (AI/ML)-based analytics on DNS queries. Machine learning models help detect and allow legitimate tunneling (some antivirus software use DNS tunneling to update endpoints) while blocking malicious tunneling intended to exfiltrate data. The AI/ML-based analytics also help detect advanced threats like domain generation algorithms (DGAs), Fast Flux, and look-alike domains.

3 Cybersecurity & Infrastructure Security Agency. (2020, November 2). *Ransomware Activity Targeting the Healthcare and Public Health Sector* | CISA. https://www.cisa.gov/uscert/ncas/alerts/aa20-302a

4 Green, A. (2020, October 19). *What is DNS Tunneling? A Detection Guide*. Varonis. https://www.varonis.com/blog/dns-tunneling

5 Palo Alto Networks. (n.d.). *What Is DNS Tunneling?* Retrieved March 9, 2022, from https://www.paloaltonetworks.com/cyberpedia/what-is-dns-tunneling

6 Roblyer, K. (2021, August 2). *3 Things NIST Taught Us About DNS Security*. BlueCat Networks. https://bluecatnetworks.com/blog/3-things-nist-taught-us-dns-security/

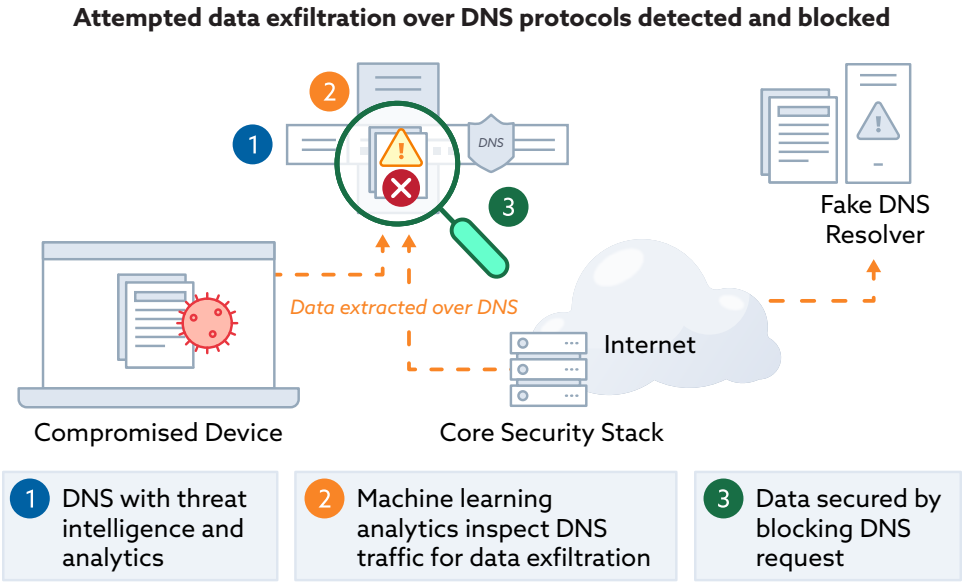

Figure 5: Detecting and Blocking DNS Data Exfiltration

1.5.3 Domain Generation Algorithms (DGAs) Control Point

Domain Generation Algorithms (DGAs) are used to generate domains so hackers can evade detection or blocking by static, domain-based uniform resource locator (URL) blocking systems (i.e., 'deny lists' on firewalls). Typically, hackers program malware to infiltrate networks and use DGAs to connect to servers controlled by the same bad actors. But instead of using static domains, the malware tries to connect to domains dynamically generated by algorithms. First, hackers will register some of the domains generated by the same algorithm and run the 'bad-actor' server (i.e., the Command and Control or CnC server) on the domain. Eventually, the malware will contact the CnC server.

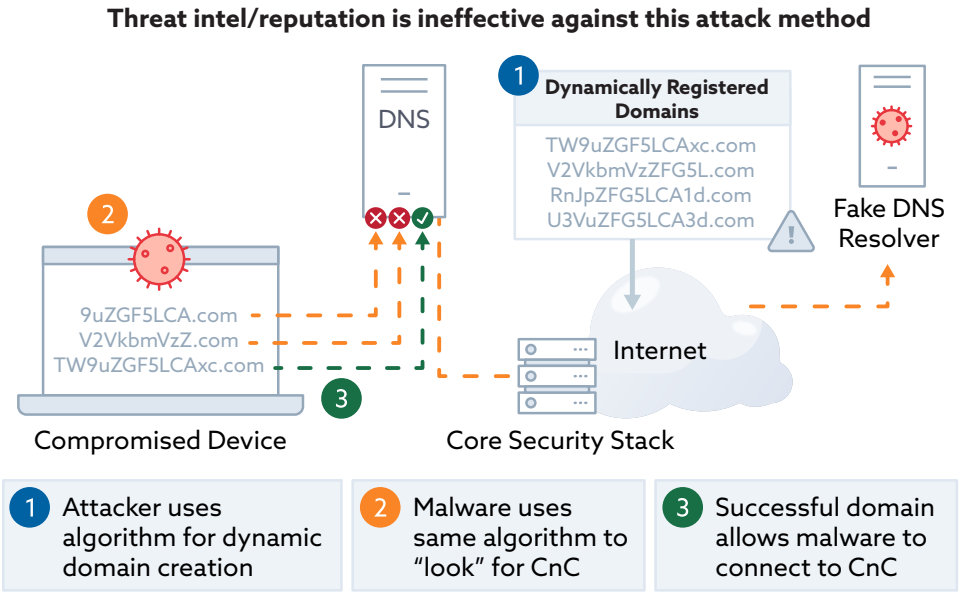

Figure 6: How DGAs Work

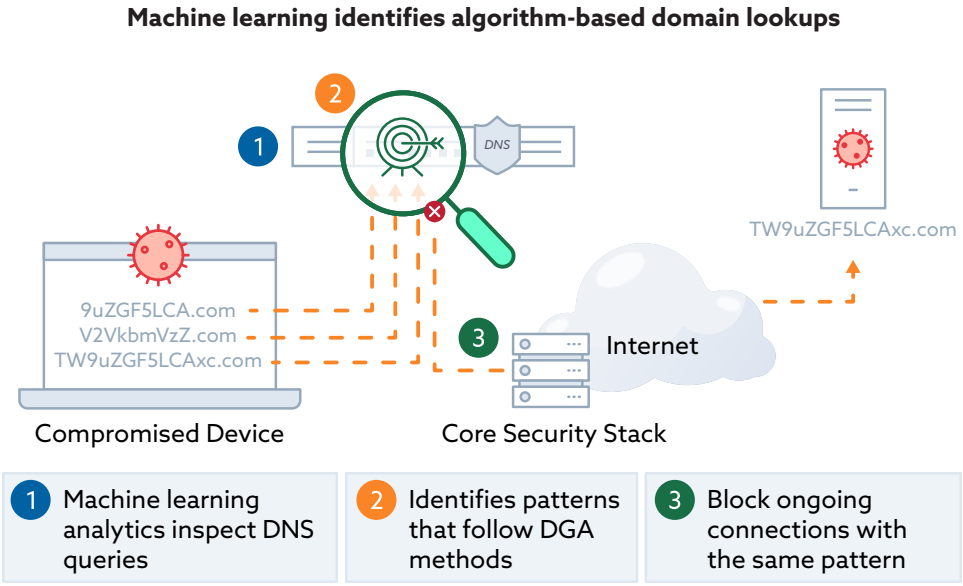

Figure 7: Detecting and Blocking DGAs

Fast flux is a DNS technique used to mask botnets by quickly shifting among a network of compromised hosts (acting as proxies). These proxies employ various Internet Protocol (IP addresses associated with a legitimate domain name, enabling cybercriminals to delay or evade detection.

Lookalike domains are fake (phishing) websites created to collect sensitive information. Examples include phony company web pages asking users to 'log in,' websites that resemble legitimate bank websites, etc.

Using AI/ML models to analyze threats like DGAs involves observing many of these threats to predict unknown versions using techniques such as entropy (level of uncertainty), linguistic analysis, frequency, and size.[7]

1.5.4 Category-Based Filtering

In addition to detecting and blocking DGA-based attacks, DNS can be a powerful network policy enforcement point for blocking access to specific content categories like social media, violence, gambling, etc. For example, a company might want to block social media access for most employees but allow it for marketing personnel who require access because accessing that is part of their job. Many enterprises have already deployed DNS filtering services for network devices to enforce these policies for users.

7 Yu, B., Pan, J., Gray, D., Hu, J., Choudhary, C., Nascimento, A. C. A., and de Cock, M. (2019, April 15) *Weakly Supervised Deep Learning for the Detection of Domain Generation Algorithms.* IEEE Access. Vol. 7, pp. 51542-51556. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8691763

1.6 Zero Trust Policy Enforcement

The National Institute of Standards and Technology Special Publication 800-207 approached Zero Trust by first describing the core components required to satisfy the needs of Zero Trust (see Figure 8 below). Zero Trust Architecture, at a high level, requires three core components before any logic can be applied to allow decisions. These include:

1. Communication: a request for an entity to access a resource and the resulting access or session.
2. Identity: the entity's identity (user or device) requesting access to the resources, with some level of authentication required.
3. Resource: any assets within the target environment.

In addition to these three core components, there are two other fundamental elements of Zero Trust:

- Policy: The governance rules that define the "who, how, what, and when" target resources are accessible.
- Data sources: The contextual information to keep policies dynamically updated.

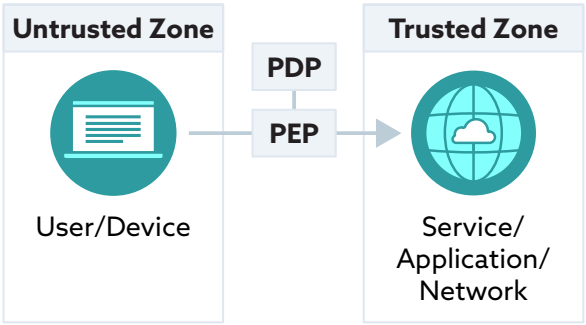

Figure 8: NIST 800-207 Zero Trust Components[8]

The policies are "rules" used to determine "who" can access "what," "when," for "how long," and "for what purpose." In the NIST ZT workflow, the policies are defined, managed, and enforced via two mechanisms:

- The Policy Decision Point (PDP).
- The Policy Enforcement Point (PEP).

Together, they regulate access to resources by being placed in the access workflow of traffic.

The Policy Decision Point (PDP) determines the "rules" applicable to each authenticated identity and communicates them to the PEP.

The Policy Enforcement Point (PEP) acts as a logical gateway to ensure that the correct access has been granted to the right entity, with the proper access levels to an approved resource.

8 Rose, S. (2020, August 11). *SP 800-207, Zero Trust Architecture* | CSRC. National Institute of Standards and Technology. https://csrc.nist.gov/publications/detail/sp/800-207/final

1.6.1 SDP and Zero Trust Policy Enforcement

The SDP architecture is designed to provide on-demand, dynamically provisioned, logically air-gapped networks.[9,10] Air-gapped networks are trusted networks that are isolated from all unsecured networks and will mitigate network-based attacks. Implementing Zero Trust requires verifying anything and everything that tries to connect to assets before granting access and the continued evaluation of sessions during the entire duration of the connection. "A Zero Trust implementation using the Software-Defined Perimeter (SDP) enables organizations to defend new variations of old attack methods that are constantly surfacing in existing network and infrastructure perimeter-centric networking models. Implementing SDP improves the security posture of businesses that face the challenge of continuously adapting to expanding attack surfaces that are increasingly more complex."[11] The enterprise must monitor the integrity and security posture of the assets (NIST SP 800-207 2020 p 7). SDP enforces this trust strategy by enabling a default drop-all firewall until users/devices are authenticated and authorized to access the assets hidden by the SDP system. In addition, by requiring pre-vetting of connections, SDP will control all connections into the trusted zone based on pre-vetting who can connect, from which devices, to what services, infrastructure, and other parameters.[12]

In its simplest form, SDP consists of Initiating Hosts, Accepting Hosts, and SDP Controllers. Initiated and accepted connection actions are managed by interactions with the SDP Controllers via a secure channel over a control plane. The SDP Controller is a policy definition, verification, and decision mechanism (a Zero Trust Policy Decision Point). It maintains information about which users/groups via which Initiating Hosts (i.e., user devices) have permission to access which organization's resources via Accepting Hosts (on-premises or in the cloud). Data is communicated over a separate secure channel in the data plane, and Accepting Hosts (typically deployed as SDP Gateways) are isolated in a trusted zone. SDP Gateways (Zero Trust Policy Enforcement Points) act as the frontend for the protected services and enforce the authentication and authorization rules maintained by the SDP Controller. Thus, in SDPs, the control plane is separated from the data plane to enable an architecturally flexible and highly scalable system.

9 For an introduction to the SDP Architecture: https://cloudsecurityalliance.org/artifacts/sdp-architecture-guide-v2/

10 For an introduction to the SDP Architecture: https://cloudsecurityalliance.org/artifacts/sdp-specification-v1-0/

11 NIST. (2020, October). *Implementing a Zero Trust Architecture.* National Institute of Standards and Technology. https://www.nccoe.nist.gov/sites/default/files/library/project-descriptions/zta-project-description-final.pdf

12 NIST. (2020, October). *Implementing a Zero Trust Architecture.* National Institute of Standards and Technology. https://www.nccoe.nist.gov/sites/default/files/library/project-descriptions/zta-project-description-final.pdf

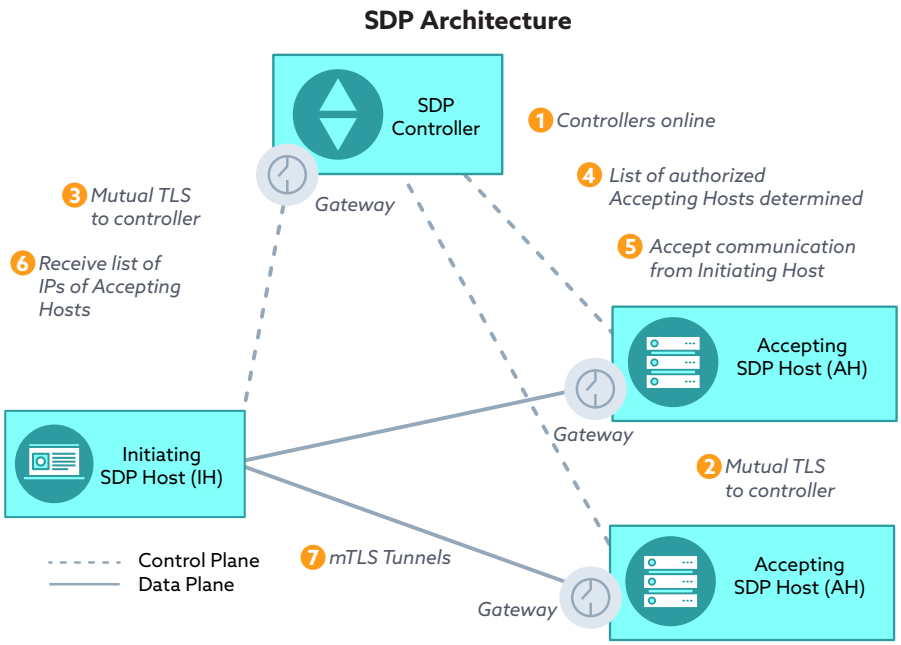

Figure 9: SDP Architecture[13]

In SDP, the Gateways are the Zero Trust Policy Enforcement Points that ensure the "rules" used to determine "who" can access to "what," "when," for "how long," and "for what purpose" are enforced.

1.6.2 DNS and Zero Trust Policy Enforcement

Zero Trust principles encourage enterprises to tie together elements of their security ecosystem, provide more context, and to better enforce access controls across Zero Trust policy enforcement points (PEPs). As introduced above, SDP is a well-proven architecture for implementing Zero Trust principles and provides identity-centric and context-aware enforcement. Thus, integrating DNS along with enterprise-managed DDI and SDP is a natural and valuable step in a Zero Trust journey, which will help enterprises get more value out of these infrastructure elements and improve the security and responsiveness of their environments. We'll explore two use cases below, illustrating how these elements connect for Zero Trust Policy Enforcement.

Policy enforcement is the application of control mechanisms to network access. Rules or policies may be based upon many criteria. DNS and enterprise-managed DDI provides essential information that can support decisions to allow user network access, and it can be a crucial component of policy enforcement.

The visibility provided by DNS and the enterprise-managed DDI solutions can send alerts to policy enforcement tools when new devices join the network. To be added to the network, new devices make a DHCP request. The stored data enables DDI to identify the device and enables tracking its activities via a "fingerprint." DDI enables mapping an IP address assignment based on the MAC

13 Cloud Security Alliance. (2019, May 7). *SDP Architecture Guide v2*. Cloud Security Alliance. https://cloudsecurityalliance.org/artifacts/sdp-architecture-guide-v2/

(Media Access Control) address and operating system as part of the basic DHCP process. If a company's device usage policy restricts access to any new device without a vulnerability scan, using the alert from the DDI solution, a NAC (Network Access Control) solution can prevent access to that device. This process can happen dynamically and in real-time.

Policy enforcement also ties directly to security objectives. For example, most basic DNS monitoring allows the more rapid discovery of malicious activity, perhaps data exfiltration attempts via DNS, and more. The security ecosystem tools such as vulnerability scanners and network access control (NAC) solutions such as SDP can also use DDI data to apply policies based on network or security events. For example, when a DNS request to a known or potentially malicious domain is detected, a scan can be triggered to check the device for vulnerabilities and malware. Or the offending device can be quarantined and be allowed to reconnect only when its vulnerabilities are mitigated. Similarly, when a Zero Trust SDP is implemented, the Policy Enforcement Point (PEP) can block connections from vulnerable devices.

2. SDP/Zero Trust and DNS Use Cases

In a Software-Defined Perimeter environment that implements Zero Trust, context is crucial to access policy decisions. Therefore, it is critical to assess risk based on identities, resources, and communications. These two use cases illustrate how DDI and SDP systems can be integrated.

2.1 Use Case #1: DNS providing Context and Metadata to SDP

This use case focuses on using the elements of DNS and enterprise-managed DDI to provide contextual information on the device and network behavior as input to the Zero Trust SDP system, enhancing its ability to make access control policy decisions.

In this use case, an enterprise is using SDP to control user access to enterprise-managed resources. Figure 10 below depicts a frequently deployed Zero Trust SDP model.[14] The key is that Zero Trust SDP uses a logically centralized Controller as a Policy Decision Point, with communication of access control policies to the SDP Gateways (Policy Enforcement Points), over a control plane. This enterprise also employs a private DDI infrastructure. In this use case, the DNS servers act as "antennae," providing additional information about devices and network activity to the SDP Controller, thus enhancing the system's ability to make access control decisions.

The model shown in Figure 10 below depicts DNS, DHCP, and IPAM (DDI) within a distributed SDP. The SDP Controller can utilize DDI data and other information to grant access, according to policy, to cloud-based and data center applications across trusted zones protected by SDP Gateways. The figure illustrates how DNS and SDP work together to provide a secure and seamless user experience. Once the SDP Controller establishes that a user device (Initiating Host, or IH) is authentic, it creates secure tunnels from the user devices to the SDP Gateways. These tunnels securely transmit both application traffic, as well as DNS requests. This enables users to access private and/or remote DNS servers without exposing them to the internet. Once a domain name is resolved, user traffic communication commences from the user device, through the Gateway, and to the target workload.

In addition to DNS resolution, the DNS server can provide device context to enhance the SDP workflow.

14 Specifically, this figure depicts the Client-to-Gateway model for clarity. The use cases in this document are equally applicable across all SDP deployment models.

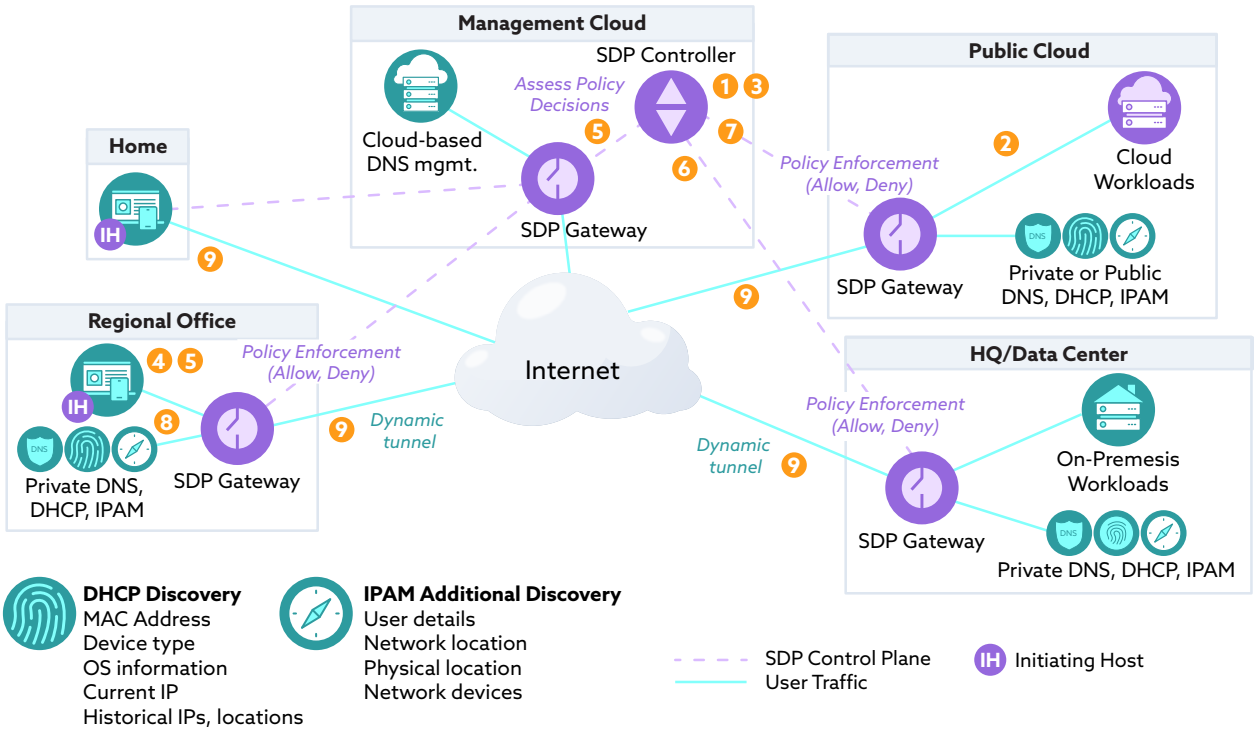

Figure 10: DNS and SDP working together

The steps portrayed in Figure 10 depict a typical sequence of events for this use case.

#	Movement	Description
1	SDP Controller Setup	One or more SDP Controllers are brought online and connected to the appropriate optional authentication and authorization services (e.g., PKI, IAM, authentication, device management, geolocation, and other such services).
2	Accepting Hosts Set up	One or more Ahs are brought online (SDP Gateways are the Ahs in the deployment model depicted). These Gateways connect to and authenticate to the Controllers. However, they do not acknowledge communication from any other Host and will not respond to any non-provisioned request.
3	Initiating Hosts Set up	One or more clients on Ihs are onboarded, and the SDP Controller authenticates each user (or Non-Person Entity). This step may include DNS resolution of the SDP Controller.
4	Initiating Host to SDP Controller	When an onboarded IH returns online (e.g., After device reboot or when the user initiates a connection), it connects to and authenticates to the SDP Controller.

#	Movement	Description
5	SDP Controller to Initiating Host	After authenticating the IH (in some cases with its corresponding identity provider), the SDP Controllers provide a list of Gateways with which the IH is authorized to communicate.
6	SDP Controller to DNS	The SDP Controller retrieves device context from the DNS server. For example, if the device is deemed malicious, the IH request is not allowed.
7	SDP Controller to SDP Gateway	The SDP Controller instructs the Gateways to accept communication from the IH and any information that defines connectivity between users, devices, and services required for two-way encrypted communications.
8	Initiating Host to SDP Gateway	The IH uses the single packet authorization (SPA) protocol to initiate a connection to each authorized SDP Gateway, which verifies the information in the SPA (for enforcement). The IH then creates a mutual TLS connection to those SDP Gateways. This step may include DNS resolution of the SDP Gateways.
9	DNS Recursive Resolution	Any IH DNS requests to resolve remote hosts behind the remote SDP Gateway are routed through the SDP tunnel to the remote private DNS server.

2.1.1 Policy Enforcement in Use Case No. 1

Access policies in a Zero Trust environment typically include information about users, devices, and the services they access.

2.1.1.1 Network Context and Identity Information

As more and more devices join the network, DNS and enterprise-managed DDI can share a comprehensive view of all devices for efficient asset management, risk reduction, and support for compliance-related policy. Additionally, each IP record managed by the DDI system may include metadata that further describes the asset and brings additional potential utility.

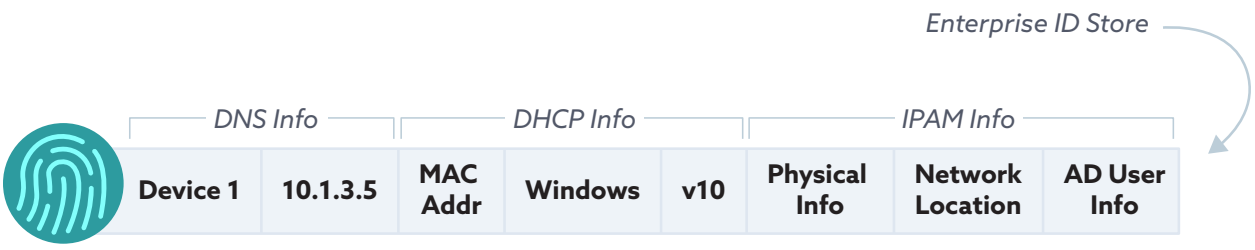

Figure 11: Context and Identity

DNS plus DHCP and IPAM data can provide additional contextual information to the SDP Controller, such as DNS requests by a given user or client device. Detailed contextual information available within DDI provides a complete fingerprint of relevant network activity that includes:

- DHCP discovery
  - MAC address
  - Operating system (OS) family type
  - OS version
  - Current IP address
  - Historical IP addresses and locations
- IPAM metadata
  - User details (through integration with IAM)
  - Subnet/network location
  - Physical location mapped by an administrator (e.g., floor, building, geolocation)

Software-defined perimeter systems can also use this contextual information to make better decisions about access. Note that private DDI systems are often unable to provide this level of information for remote users since those users are no longer directly connected to enterprise networks. Often, remote access solutions such as VPNs can obfuscate users and devices from on-premises enterprise DDI systems (i.e., mapping all remote users to a single shared private IP address, or NAT). Software-defined perimeter systems will compensate for this by providing uniformly deep context and information about users and devices (regardless of location). While only local users will utilize the DHCP aspect of DDI, both on-net (on the enterprise network) and off-net (e.g., at home) users will implement DDI systems for DNS requests. Therefore, DDI can provide information about these requests to the SDP system.

The activity of the two following users is depicted in Figure 12 below.

- Natalia works in the office and uses the enterprise-controlled DHCP server on the local area network (LAN) to obtain an IP address. She also uses the local DNS server for all her DNS requests, giving visibility into her DNS activity.
- Jim works from home and uses his local network router for DHCP, meaning the enterprise DDI system doesn't have visibility into his DHCP request or fingerprint. However, the SDP system ensures his DNS requests for enterprise resources are sent across the SDP tunnel and resolved by the enterprise DNS server—providing complete visibility into his DNS activity. Note that the SDP implementation may tunnel all or some of Jim's DNS traffic; generally, this is configurable.

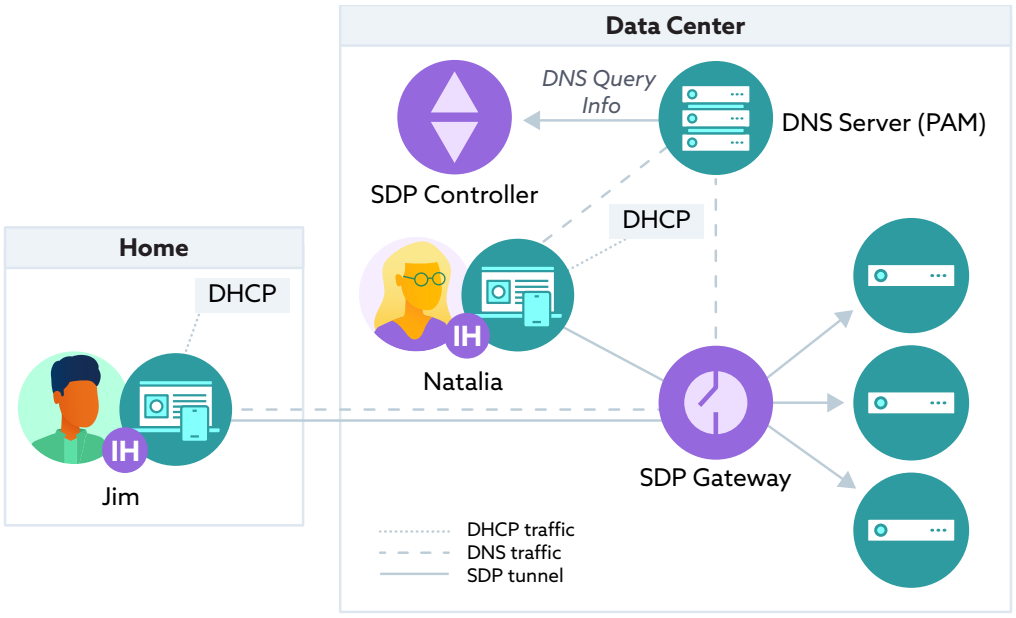

Figure 12: On-Net and Off-Net User Traffic

An SDP implementation can leverage network context and identity data to define policies for the SDP Controller and enforce them at the SDP Gateways. The SDP Controller can integrate many information sources—such as identity providers (directories) and vulnerability scanners—to provide enforceable fine-grained policies.

2.1.2 Responding to Malicious Activity

As shown in figure 12, private DNS servers are often the first point where malicious activity (or an indication of compromise). One example is a DNS request to a CnC server associated with ransomware after an initial endpoint infection.

The DNS is a highly effective way to attribute activity to a specific device. Additionally, DDI's detection capabilities can be used to trigger a response by SDP through a combination of device network access modifications, enhanced authentication requirements, and notifying users to take specific actions. For example, SDP can help disambiguate network activity if many remote users map to a single, shared IP address on a local network (e.g., source NAT configuration) for network traffic from an SDP Gateway to a protected resource.

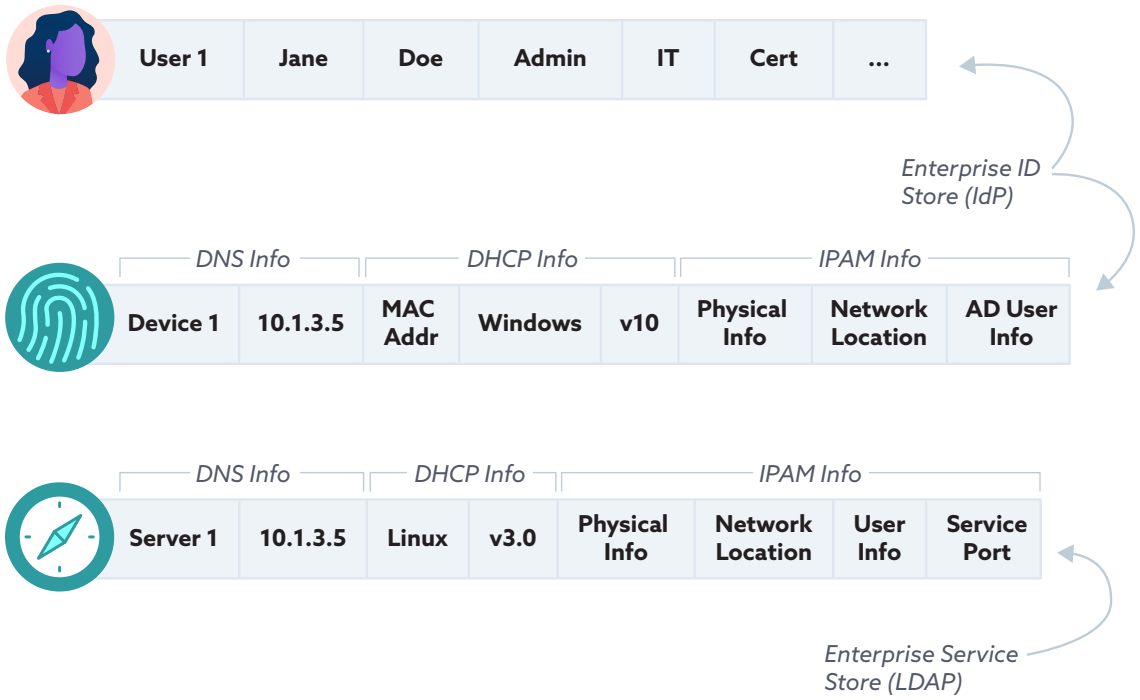

Figure 13: Using DNS, DHCP and IPAM Information to Pinpoint Infected Devices

2.1.3 Location-Based Access Controls

Access may or may not be granted based upon the geographic location of a user/device, including which part of the network, switch port, or wireless access point the user is connected to. For example, a medical clinic where vitals are taken using a rolling cart could enact a policy stating those devices will only be granted network access from the clinical side of the building and not the front office. Network segmentation is key to securing resources and limiting attack risks and infiltration impacts. In this scenario, IPAM metadata such as network and physical location could inform the SDP Controller of a device's whereabouts.

2.1.4 Device-Based Access Controls

Zero Trust is about knowing as much as possible about the device at the earliest possible moment to establish trust. Therefore, it is critical to extract information from the DHCP process and gather device data. Using techniques like DHCP fingerprinting (identifies the device requesting a DHCP lease when trying to retrieve an IP address) and MAC analysis, there is rich device metadata available, including:

- Hardware vendor
- Class of device (phone/tablet/PC/IoT)
- Operating system and version
- MAC address
- IP address

The SDP Controller can use this information to determine access, especially as the common practice of "Bring Your Own Device" (BYOD) continues. Allowing these devices to access network resources means ensuring they do so securely. For instance, an iPhone running the latest iOS version may present sufficiently low risks for network access, while an older Android phone running an outdated version may mount a high enough risk of malware to warrant blocking or limited network access.

This device context is essential for the Zero Trust Policy Decision Point (SDP Controller), as the SDP Controller evaluates policies to determine whether they apply to a given subject. The SDP implementations capture this type of information from the local user agent (SDP Client).

The device metadata should be included in the policy assignment criteria for sound Zero Trust deployment.

2.1.5 User-Based Access Control

The preceding scenario may expand to allow only certain users to access specific applications. For example, by integrating with an enterprise directory, user information and DNS-provided IP address information can be integrated into the SDP system as additional context or an additional integrity check for access control policies.

2.2 Use Case #2 - SDP Controller Publishing Policy Decisions to DNS

The other use case where DNS and the SDP Controller can improve security together involves the controller publishing access policy decisions to the local DNS server, for an additional layer of enforcement. For example, let's say only authenticated users in the Finance group should be able to access the `finance.internal.company.com` web app. In contrast, only Engineering users should be able to access the `git.internal.company.com` server. All employees should be able to access the `email.internal.company.com` server, and all devices should have DNS queries for external domains (e.g., `cloudsecurityalliance.org`) permitted, as depicted in figure 14 below.

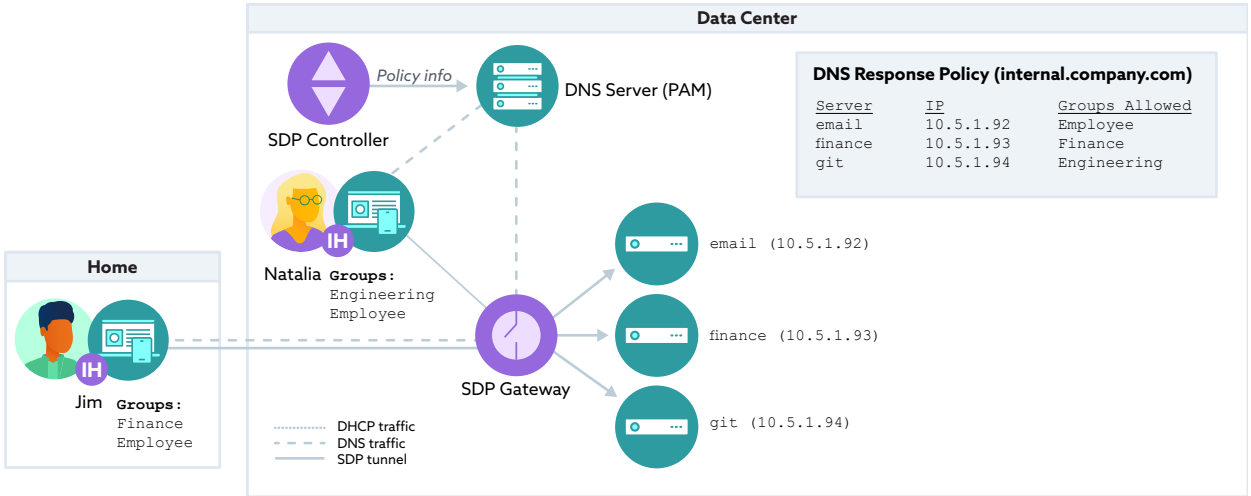

Figure 14: DNS Resolution Using SDP Policies

Figure 14 shows how the SDP Controller has pushed policy information to the DNS server, showing which directory groups should be permitted to resolve the hostnames for which internal servers. (Note that this diagram assumes the DNS server has obtained the user-to-group mapping from an enterprise directory service, not shown. Alternatively, the SDP Controller could provide this information to the DNS server). The remote user, Jim, has his DNS requests for internal resources tunneled through the SDP Gateway to the private DNS server. The on-premises user, Natalie, has her DNS requests routed directly to the same DNS server, which is local and on-network. In both cases, the DNS server can distinguish which user is making the DNS request, and also knows to which groups each user belongs. The SDP Controller has published policy information to the DNS server, indicating which directory group memberships are required for access to which services, as shown in the DNS Response Policy table in the diagram. Based on all this information, the DNS server will return an IP address in response to a query only if the requesting user is in a matching group. So our user Jim will be able to obtain the IP address for the `email` and `finance` servers, but not for the `git` server.

2.2.1 Policy Enforcement in DNS - an additional layer of security

DNS access control is a simple, scalable, and cost-effective method to secure applications and data in a Zero Trust/SDP environment. Detecting and blocking malware communications, and doing category based filtering to block access to certain categories of content (such as social media, violence, gambling etc.) at the DNS level can significantly reduce the amount of malicious traffic to NGFWs and gateways. This offloading of threat-s preserves the processing power of firewalls and gateways, which are often expensive perimeter solutions, and improves the ROI of those solutions.

Note that this should be viewed as a control that may be deployed in addition to network-based enforcement via an SDP Gateway for the trusted enclave. Preventing DNS resolution of a target server, as shown above, in this use case, impedes unauthorized access but is not a substitute for inline enforcement. For example, an attacker could create a `hosts` file entry with the target server's IP address, reducing the effectiveness of DNS-based filtering. Or, an attacker could simply enumerate IP addresses without performing DNS resolution on their hostnames.

3. Conclusion

This whitepaper explored how enterprise DDI systems can augment and integrate with SDP to enhance an organizations' security, resiliency, and responsiveness. Enterprise DDI can help improve security by filtering known bad sites and detecting Indicators of Compromise. SDP provides rich contextual information for enforcing security policies, and both of these systems can benefit by being integrated. DNS can provide enhanced contextual device and activity information to the SDP Controller for better policy decision-making. And DNS systems can consume Zero Trust context and decisions from the SDP CController, extending the reach of SDP, and effectively making enterprise DNS a focused Zero Trust Policy Enforcement Point.

Information security will always be multi-layered — Defense in Depth is a crucial concept, and Zero Trust via SDP is an approach that benefits from integration with many other parts of an enterprise security infrastructure. Enterprise DNS is one example of this, illustrated by the two use cases explored above.

4. References

Cloud Security Alliance. (2014, April 30). *Software-Defined Perimeter (SDP) Specification v1.0.* Cloud Security Alliance. https://cloudsecurityalliance.org/artifacts/sdp-specification-v1-0/

Cloud Security Alliance. (2019, May 7). *SDP Architecture Guide v2.* Cloud Security Alliance. https://cloudsecurityalliance.org/artifacts/sdp-architecture-guide-v2/

Cloud Security Alliance. (2019, October 27). *Software-Defined Perimeter as a DDoS Defense Mechanism.* Cloud Security Alliance. https://cloudsecurityalliance.org/artifacts/software-defined-perimeter-as-a-ddos-prevention-mechanism/

Cybersecurity & Infrastructure Security Agency. (2020, November 2). *Ransomware Activity Targeting the Healthcare and Public Health Sector* | CISA. https://www.cisa.gov/uscert/ncas/alerts/aa20-302a

Garbis, J., and Chapman, J. W. (2021) *Zero Trust Security: An Enterprise Guide.* Apress. https://www.apress.com/us/book/9781484267011

Green, A. (2020, October 19). *What is DNS Tunneling? A Detection Guide.* Varonis. https://www.varonis.com/blog/dns-tunneling

Infoblox. (2019) *Using Artificial Intelligence/Machine Learning to Detect Domain Generation Algorithms.* Infoblox. https://info.infoblox.com/resources-whitepapers-artificial-intelligence-to-detect-domain-generation-algorithms

Infoblox. (n.d.). *Infoblox Glossary*. Infoblox. https://www.infoblox.com/glossary/

Infoblox. (2019, January). *Ryuk Ransomware Cyber Report.* Infoblox. https://insights.infoblox.com/threat-intelligence-reports/threat-intelligence--3

Infoblox. (2019). *Powering SOAR Solutions from the Foundation*. Infoblox. https://insights.infoblox.com/solution-notes/infoblox-solution-note-powering-soar-solutions-from-the-foundation

Johnson, D. (2021, February 16). *What is a DNS server? How Domain Name System servers connect you to the internet.* Business Insider. Retrieved March 9, 2022, from https://www.businessinsider.com/what-is-a-dns-server?r=US&IR=T

NIST. (2020, October). I*mplementing a Zero Trust Architecture. National Institute of Standards and Technology.* https://www.nccoe.nist.gov/sites/default/files/library/project-descriptions/zta-project-description-final.pdf

OmniSecu.com. (n.d.). *Recursive and Iterative DNS Queries.* Retrieved March 9, 2022, from https://www.omnisecu.com/tcpip/recursive-and-iterative-dns-queries.php

Palo Alto Networks. (n.d.). *What Is DNS Tunneling?* Retrieved March 9, 2022, from https://www.paloaltonetworks.com/cyberpedia/what-is-dns-tunneling

Roblyer, K. (2021, August 2). *3 Things NIST Taught Us About DNS Security.* BlueCat Networks. https://bluecatnetworks.com/blog/3-things-nist-taught-us-dns-security

Rose, S. (2020, August 11). *SP 800–207, Zero Trust Architecture* | CSRC. National Institute of Standards and Technology. https://csrc.nist.gov/publications/detail/sp/800-207/final

Waverley Labs, SDP Center, *Open Source Reference Implementation* (funded by DHS), 2021, available at http://sdpcenter.com/test-sdp/

Yu, B., Pan, J., Gray, D., Hu, J., Choudhary, C., Nascimento, A. C. A., and de Cock, M. (2019, April 15) *Weakly Supervised Deep Learning for the Detection of Domain Generation Algorithms.* IEEE Access. Vol. 7, pp. 51542-51556. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8691763

5. Acronyms

AI/ML: Artificial Intelligence/Machine Learning

BYOD: Bring Your Own Device

DDI: DNS, DHCP, IPAM

DGA: Domain Generation Algorithms

DHCP: Dynamic Host Configuration Protocol

DNS: Domain Name System

DDoS: Distributed-Denial-of-Service

EDR: Endpoint Detection and Response

IaaS: Infrastructures-as-a-Service

IP: Internet Protocol

IPAM: Internet Protocol Address Management

ISP: Internet Service Provider

MAC: Media Access Control

NAC: Network Access Control

PaaS: Platform-as-a-Service

PEPS: Policy Enforcement Points

RPZ: Response Policy Zone

SDP: Software-Defined Perimeter

SIEM: Security Information and Event Management

SOAR: Security Orchestration, Automation, and Response

TLD: Top Level Domain

ZTNA: Zero Trust Network Access