Technical Report
1253

Zero Trust (ZT) Concepts for Federal Government Architectures

K.D. Uttecht

30 July 2020

Lincoln Laboratory
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LEXINGTON, MASSACHUSETTS

This material is based upon work supported by the Department of Homeland Security under Air Force Contract No. FA8702-15-D-0001.

This report is the result of studies performed at Lincoln Laboratory, a federally funded research and development center operated by Massachusetts Institute of Technology. This material is based upon work supported by the Department of Homeland Security under Air Force Contract No. FA8702-15-D-0001. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the Department of Homeland Security.

Massachusetts Institute of Technology
Lincoln Laboratory

Zero Trust (ZT) Concepts for Federal Government Architectures

K.D. Uttecht

Group 57

Technical Report 1253

30 July 2020

Lexington Massachusetts

This page intentionally left blank.

ACKNOWLEDGMENTS

Main Author: Karen Uttecht

Contributing Technical Team: Paula Donovan, Kevin Perry, Sandeep Pisharody, Mary Ellen Zurko

Technical Editor: Judy Marchese

Department of Homeland Security (DHS) Science and Technology Directorate (S&T) Program Manager: Alex Karr

Contributor Contact Information:

g57-office@ll.mit.edu

{karen.uttecht, pjdonovan, kevin.perry, sandeep.pisharody, maryellen.zurko}@ll.mit.edu

alex.karr@hq.dhs.gov

This page intentionally left blank.

EXECUTIVE SUMMARY

This report describes the concept of Zero Trust (ZT), based on the key idea that an organization should not implicitly trust any network traffic, device, or user solely based on their physical or logical network location. Instead, ZT focuses on protecting resources. It requires any and all communication to be between explicitly verified and authorized users and devices. Further, any and all communication should be monitored. ZT is often misrepresented as eliminating firewalls; it is more accurate to say ZT places firewall-like policy enforcement points throughout the network. This eliminates the traditional firewall as a gateway from outside to inside, but still provides the same filtering of traffic.

The core principles behind ZT are: 1) Universal authentication of all users, devices, and services; 2) Access segmentation, allowing no single entity access to more than a small portion of the organization's resources; 3) Minimal trust authorization, keeping access to resources only to those entities that "need-to-know" and can be trusted; 4) Encryption everywhere to protect information in flight and at rest, whether inside or outside the organization's networks; and 5) Continuous monitoring and adjustment to detect issues early and adjust access accordingly.

Despite its popularity, realizing the concept of ZT has some critical shortcomings. Neither a universally agreed-upon definition of what exactly makes up ZT, nor a set of criteria on when ZT is implemented properly exist. It is believed the concepts behind ZT will confer a greater level of security, but there is no research proving the degree of benefit these principles or capabilities actually provide. ZT also needs organization-wide buy-in to implement burdensome restrictions successfully.

As a result, there are many challenges in implementing ZT on federal networks. Choosing technologies is difficult due to the wide range of different product choices, as well as a lack of independent analysis into their effectiveness. Vendor's proprietary interfaces prevent integrating capabilities. No metrics for measuring success exist. Government organizations may have conflicts between ZT principles and policies to which they must adhere. The scale of the data to be monitored and analyzed is difficult to manage and beyond the capabilities of many current solutions.

This report provides a breakdown of the dimensions and capabilities that make up a ZT architecture. The three National Institute of Standards and Technology (NIST) models are summarized, and case studies of four ZT architectures are described: BeyondCorp, Next-Generation Firewall (NGFW)/Forrester, Software-Defined Perimeter, and VMWare/NSX. Of these choices, the architecture that is the best fit for a particular organization depends on the mission of that organization. For a public service agency whose main mission is to interact with the public, any of the architectures would meet their needs, and which to choose depends heavily on the existing infrastructure. A virtualized architecture similar to the VMWare/NSX would provide the most benefit, though it may be the most difficult to migrate to. For a public safety-focused agency with many field agents who do not have continuous internet access and utilize many sensors and Internet of Things (IoT) devices, an SDP-based solution is recommended. For a larger umbrella organization that may have sub-organizations, as the previous two exemplars, a federated architecture is recommended to be able to accommodate the diversity of requirements.

Organizations can choose between a single solution or federate several solutions. Federation is a better choice for larger organizations, due to the diversity in their missions and technologies supported. Several approaches for implementing a ZT architecture are discussed, but they all rely on a staged rollout where technology is deployed first in audit-mode and policy issues can be ironed out before full enforcement is turned on.

Key best practices for implementing a ZT architecture include: prioritizing security from senior leadership down to avoid exceptions undermining ZT principles; collaborating to prioritize capability roll-out while educating the organization on their vulnerabilities and how to fix them; utilizing existing software and capabilities to implement the principles of ZT reducing costs; investing in automation to make policy and user-access granting easy to both give and revoke keeping ZT posture strong and avoiding granting unnecessary privileges for longer than needed; contingency planning for "break the glass" solutions to mitigate the risk of losing access to critical resources under stringent ZT access restrictions; lastly, rolling out policies and access restrictions incrementally, allowing the organization to learn from early mistakes.

TABLE OF CONTENTS

This page intentionally left blank.

LIST OF FIGURES

This page intentionally left blank.

LIST OF TABLES

This page intentionally left blank.

1. INTRODUCTION

Zero Trust (ZT) is a cybersecurity framework built upon the key idea that an organization should not implicitly trust any network traffic, device, or user. Everything must be explicitly verified, authorized, and monitored.

1.1 WHAT IS "ZERO TRUST?"

The term Zero Trust itself is relatively new, but it bundles together particular existing best security practices, with an emphasis on their importance and the extent to which they need to be utilized.

Despite its name, complete z*ero trust* is not truly possible. Instead, what is really meant is no trust without verification. Further, the trust given to any entity approaches as close to zero as possible while still allowing that organization's mission to succeed. The core purpose of ZT is to structure the organization's information technology (IT) infrastructure in such a way that should an adversary succeed at gaining access to one device or a user's credentials, their ability to move through the network to reach their target is severely inhibited, thus providing defense-in-depth.

1.1.1 Core ZT Principles

- **Universal Authentication:** Any entity (device, user, application, etc.) that operates within or interacts externally with the organization's IT infrastructure is authenticated. This includes user and device authentication not just externally but also on internal networks.

- **Access Segmentation:** All access, whether network access, data access, application access, etc., should be segmented into the smallest accessible pieces possible so that no single entity (device, user, application, etc.) has access to the whole or a large part of the organization's network, data, or applications, and so that the fewest possible entities have access to critical data. This especially applies to administrator access, segmenting accounts with ability to change critical controls so they cannot access anything else.

- **Minimal Trust Authorization:** Hand-in-hand with access segmentation is extending the minimum amount of trust possible. This means restricting entities to the very least amount of privilege (access, administration rights, etc.) necessary for that entity to perform its mission. Not only does this include minimizing what can be accessed, but also for how long it can be accessed, and where it can be accessed from. Additionally, evaluating the trustworthiness of the entity and not authorizing entities whose context or attributes indicate they should not be trusted.

- **Encryption Everywhere:** ZT assumes that the network cannot be trusted. This means that anywhere communication is happening, it is assumed an adversary can be monitoring. So, all communications should be encrypted from end-to-end, including on internal organization networks.

- **Continuous Monitoring and Adjustment:** All entities operating on the organization's infrastructure should be monitored. This includes all network traffic, access attempts and successes, and software being run. Monitoring should not just be for malware signatures, but on a much larger scale, including all network and system events, (e.g., attempts to access anything outside of what is authorized within the organization) as well as for cross-checking data from different sources. This information should then be used to adjust the organization's policies.

Together, these five principles make up the concept of ZT. They should be applied at many different levels and domains, from the network to cloud and other applications, for both users and administrators. The main goal of ZT is to disrupt the cyber kill chain at the point where the attacker is attempting to move laterally through the organization to reach the intended target. ZT principles do this in two ways. First, eavesdropping and reconnaissance is difficult because end-to-end encryption and universal authentication limit access to information, and constant monitoring identifies abnormal communications. Second, lateral movement is reduced because segmentation of access and least privilege authorization prevent compromised entities from having access to critical resources. When lateral movement is restricted, insider threats are also limited, because an insider threat and an initial compromise are very similar, i.e., the adversary already established access, but needs to move latterly within the organization's systems in order to access or affect the critical resource to accomplish their goal.

1.2 BACKGROUND

"Zero Trust Networking" was introduced in 2009 by John Kindervag of Forrester Research. Initially it was very focused on the network itself [1]. In 2010, Google underwent the first documented nation-state sponsored cyberattack, dubbed "Operational Aurora" [2]. As a result, Google embarked on a six-year project they called BeyondCorp, a company-wide initiative to completely re-invent security from the ground up in a Zero Trust (ZT) model [3]. The BeyondCorp model used the core principles of ZT Networking differently, applying them directly to Google applications. They published their architecture in 2014 [4]. In 2018, Chase Cunningham at Forrester Research published a follow-on to the original ZT paper detailing what they called "The Zero Trust eXtended (ZTX) Ecosystem" expanding ZT Network concepts to encompass data, people, devices, and workloads [5].

The success of BeyondCorp as a real-world implemented ZT system launched the concept of ZT into the limelight. Following the Office of Personnel Management (OPM) breach in 2015, the U.S. House of Representatives Committee on Oversight and Government Reform issued a report recommending federal information security efforts be reprioritized toward a ZT Model [6]. In the following years, the number of products marketed as "Zero Trust" increased significantly. However, the term is often just a marketing buzzword, and not one that comes with a clear definition. This paper attempts to provide a view of the concept of ZT—independent of a particular product or marketing firm—with an emphasis of how to apply ZT concepts within federal government architectures.

Attempts are being made to provide a common understanding of ZT apart from commercial interests. In September 2019, the National Institute of Standards and Technology (NIST) released a draft special publication "SP800-207 Zero Trust Architecture," which describes the core components that make up a ZT Architecture, with an emphasis on network architectures. A second draft for additional comments was released in February 2020 [7].

1.3 SHORTCOMINGS

ZT has gained momentum in recent years, with the market for ZT products expected to double by 2024 [8]. Although there is not a lot of critical literature on the topic, there are some critical shortcomings discussed in this section.

1.3.1 Lack of Definition and Completion Criteria

The concepts that make up ZT are not new and many are standard in industry best practices. However, there is not an industry standard on how these concepts should be applied to produce an implementation of ZT. How does an organization know when it has met its goal of implementing ZT? Certainly, if an organization has gaps in their implementation of any of the key concepts outlined above, those need to be covered. But other gaps are a matter of degree: how much network segmentation is needed? How much do we need to restrict user roles and application access? What metrics can be evaluated to validate or determine the effectiveness of implementations? As a result of a lack of clear, consistent guidance in these areas, it is challenging for an organization to have a clear idea if it has successfully implemented ZT and very easy for a company to claim its product provides it.

1.3.2 Degree of Benefit

The concepts at the core of ZT are beneficial to an organization based at best on observational evidence, not formal proof. Additionally, quantitative measurements of an organization's security posture are lacking and therefore it is difficult to have a reliable, measurable, repeatable way to quantitatively verify the change in security posture provided by implementation of various aspects of ZT. Organizational budgets are tight, and therefore it is important to know whether spending millions of dollars to pursue a more granular segmentation implementation would provide any benefit over a simpler, less costly option.

1.3.3 Requires Security to be an Organization-Wide Priority

Implementing the concepts of ZT may involve a considerable cost in terms of productivity and re-training for users and IT administrators. This cannot be done properly if an organization isn't willing to prioritize security controls over easy availability, especially for senior leadership.

ZT is about only allowing access to a minimal amount of resources and only in exacting circumstances. This cannot work if upper management demands they have the ability to access everything, or that security be overridden for them. It cannot work if security procedures are not enforced or are constantly overridden. Similarly, IT administrators have to make security a priority as part of their job, as they are the ones who will be most impacted, as the group with the most access to critical resources.

1.4 MISCONCEPTION – NO MORE FIREWALLS

One of the phrases that is constantly used in connection with ZT is "No more firewalls!"—that an organization is somehow doing away with the concept of firewalls. This idea is inaccurate, and perhaps used to garner attention. It is important to understand that ZT does not do away with the concept of a firewall—as defined as a network security system that monitors and controls network traffic. What is being referenced is moving away from the idea that the *only* place this is being done is at the network ingress/egress point (also known as the castle-and-moat model of security). Instead, a better way to describe ZT would be to say "Firewalls everywhere!"—not just at the ingress/egress point, but at many points within the network as well—segmenting the network into many different smaller networks, access to which is controlled by the various firewalls. This is sometimes described as segmentation, micro-segmentation, or micro-perimeter and is a good example of implementing defense-in-depth. Passing the firewall to enter the network is not a key to access anything—from an authorization standpoint there is no inside or outside—everything must be authenticated and authorized.

1.5 TERMINOLOGY

Since ZT is a newer concept and various vendors have their own branded terms they prefer to use, the terminology used by the overall ZT community has not unified. This document will adhere to the terminology used by NIST in its ZT Architecture special publication (currently in draft status), as it has created a vendor-neutral phraseology. In order to make things as clear as possible, a glossary has been included at the end of this document.

2. ZERO TRUST (ZT) ARCHITECTURES

The popularity, yet non-standardization of ZT means everyone has their own concepts of what it is and what it entails. While these systems are described as "Zero Trust Architectures," they are not necessarily all encompassing. The concepts of ZT should not just be deployed at one level or in one area of the enterprise. They need to be applied pervasively to provide defense in depth. This means applying concepts at very low levels of the TCP/IP networking stack like the Ethernet or IP network layer, through to higher levels like the application layer, and beyond to the permissions within applications themselves. They need to be applied not only within the bounds of the enterprise buildings but also to the cloud and users operating remotely.

2.1 DIMENSIONS OF ZERO TRUST (ZT) TECHNOLOGY

When comparing ZT technologies, it is important to note that they can differ in several dimensions, and that some architectures may operate at one of these levels or several. Generally, technologies are focused at a specific level of the dimensions described in the following sections.

2.1.1 Network Protocol Layer

The tenets of ZT can be applied to many different parts of the enterprise infrastructure, but the main focus of many ZT architectures is still on securing the network, due to the origins of ZT as a network security solution. It is critical to have a Policy Enforcement Point (PEP) on the network controlling connection requests and packet flows, but choosing which layer of the network protocol stack to implement enforcement on can result in very different solutions. ZT architectures can also work on multiple layers to provide defense-in-depth.

- **Data Link** – Authentication of physical/Media Access Control (MAC) addresses before allowing access to the network
- **Network Layer** – Policy enforcement on network layer devices such as routers, firewalls, and switches enforce rules based on IP address, or by adding an overlay of encrypted tunnels and Virtual Private Networks (VPNs)
- **Transport** – Enforcing rules on specific TCP/User Datagram Protocol (UDP) ports and protocols, as well as some types of overlays/VPNs may operate at this level
- **Application** – Enforcing who can access specific applications and potentially specific parts of an application, for example enforcing which Uniform Resource Locators (URLs) on a web-based application a particular end-point can access

Policy enforcement at multiple networking stack layers provides defense in depth, though some combinations will have diminishing returns. Layering a solution that creates an overlay network of Internet Protocol Security (IPSec) tunnels for example, which encrypt and hide the transport layer addressing, would not be improved by trying to enforce transport layer rules on a router, as it could not see what transport protocol or port the traffic was addressed to.

2.1.2 Policy Enforcement Point (PEP) Location

Policy enforcement can be done at many locations within the enterprise architecture. The priorities and budget of the organization and the infrastructure already in place may make one location preferable to another. Multiple enforcement locations can also be combined to provide defense in depth.

- **Centralized** – A single or set of routers or next-generation firewall can enforce policy on traffic passing through.
- **Enclaved/Gateway** – Enforcement is done at the entry point of an enclave of several resources or devices by a gateway that inspects traffic.
- **Proxied** – Enforcement is done by a proxy application, which then passes traffic on to the application/resource being accessed.
- **Application Enforcement** – Enforcement distributed throughout the network by integrating the enforcement point with the application itself.
- **Hybrid** – A hybrid version of any of the above models, for example having some applications proxied while others have integrated application enforcement. Generally, the choice of a Proxied or Application Enforcement approach will require some amount of hybrid support for legacy applications and resources that cannot be integrated, either for technology or budget reasons.

Generally, centralized enforcement is less expensive to deploy because it does not require integration with resources and the cost to users is lower because the deployment is generally at a lower network level and less visible to users. However, this also means there is less contextual information available to use to enforce policy, since this information is generally known at the endpoints and not available in the network traffic.

On the other end of the spectrum, enforcement in the applications provides the nuanced and segmented policy enforcement, but requires significant outlay to integrate the system with the application or resource. Users may have to make changes to their workflow or behavior as a result of the new system, or new software or technology will need to be deployed on their endpoint devices. When legacy devices must be supported, this indicates a need for a PEP location other than Application Enforcement. If there are many legacy devices, this may be a driver for the organization to choose a Centralized, Enclaved/Gateway, or Proxied approach, in order to avoid having to support a hybrid architecture, which would require federation of multiple ZT technologies.

2.1.3 Domain

While ZT was originally conceived as a network security model, the ZT tenets can be applied to other domains within an enterprise system as well. Many technologies focus on one of these areas, so as to provide a comprehensive ZT architecture. In this case, different solutions may need to be used for each area. While not an exhaustive list, Table 1 describes how each of the ZT principles described in Section 1.1 can be applied to different domains within the enterprise network.

Table 1

How Zero Trust (ZT) Concepts Can Apply to Different Domains of the Enterprise Network

Domain	Principle	How ZT Principle can be Applied
Network	*Universal Authentication*	Only authenticated users and devices can access the network.
	Access Segmentation	Portions of the network are segmented and only authorized traffic allowed.
	Minimal Trust Authorization	Network traffic is whitelisted to only necessary connections.
	Encryption	All communications are encrypted, even on internal Local Area Networks (LANs).
	Monitoring and Adjustment	All connections, traffic, and unauthorized connection attempts monitored. Behavior indicating potential compromise disables network access.
Device	*Universal Authentication*	Devices are password/2FA protected and authentication is done on devices and running software.
	Access Segmentation	Elevated privileges are restricted and time-limited.
	Minimal Trust Authorization	Application use is restricted to whitelisted applications. Applications are sandboxed.
	Encryption	Data on hard drive is encrypted.
	Monitoring and Adjustment	Login attempts, software installed, device location, and attempted connections are monitored. Detected compromise, unpatched software, etc. result in reduced access or lockout.
Application/ Workloads	*Universal Authentication*	Only authenticated users on authenticated devices can access resource, application, or workload.

Domain	Principle	How ZT Principle can be Applied
	Access Segmentation	Application access/functions and workloads are divided into segments. Access allowed to users with specific roles.
	Minimal Trust Authorization	Users are given only the roles they need to complete work and roles are time-limited or re-evaluated frequently.
	Encryption	Data at rest is encrypted.
	Monitoring and Adjustment	Access attempts, actions taken, roles granted, and privilege escalations are monitored. Suspicious behavior and access attempts result in reduced access or lockout.
Cloud	*Universal Authentication*	Authentication of all users and devices connecting.
	Access Segmentation	Segment by roles similar to App/workloads. Administration functions should be segmented so one account does not have full control of all cloud functions.
	Minimal Trust Authorization	Users are given only the roles they need; administrative functions are given for only limited time and only to those admins who require it.
	Encryption	Data at rest, data passed between applications and workloads, and all communications to/from/within the cloud are encrypted.
	Monitoring and Adjustment	Access attempts, actions taken, roles granted, privilege escalations, as well as changes to the cloud provider are monitored. Suspicious behavior and access attempts result in reduced access or lockout.

2.2 CAPABILITIES

In order for a ZT architecture to implement the core principles of ZT, it must utilize a number of different capabilities. In the market, sometimes several of these capabilities are available as a single integrated system. Figure 1 shows a notional enterprise IT system architecture with examples of components that make up a ZT solution.

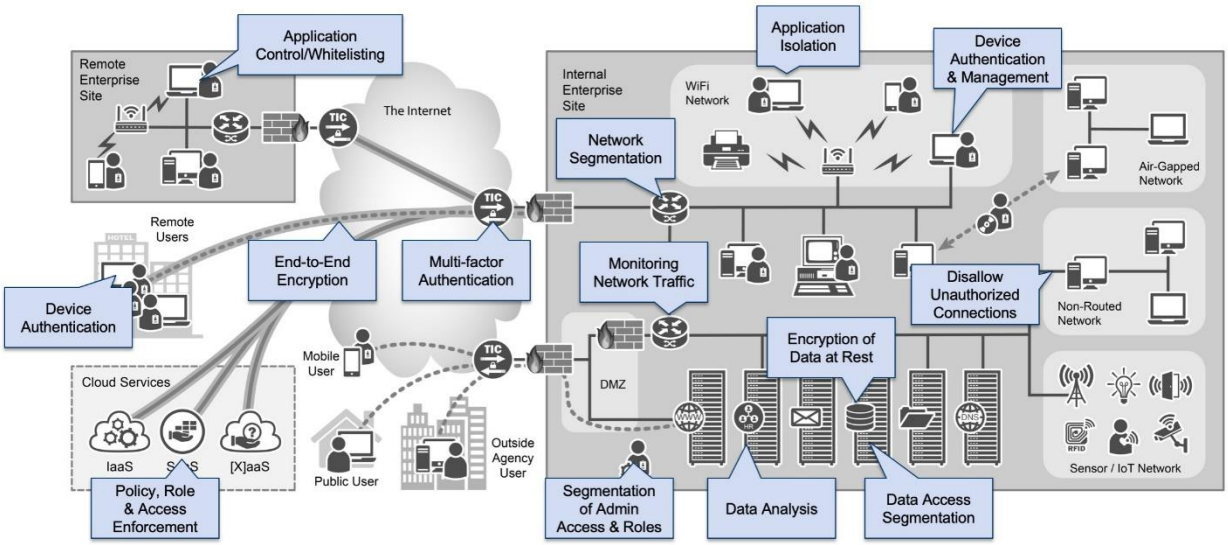

Figure 1. Capabilities that make up a ZT architecture shown on a notional enterprise.

The following are categories of capabilities that combine to form a ZT architecture:

- **Network traffic filtering** – using firewall technology throughout the network to enforce network segmentation and prevent unauthorized connections, monitoring traffic for disallowed or unusual behavior
- **Network access control** – network segmentation, authenticating devices before allowing access, requiring user authentication for network access
- **Local system access control** – disk encryption, cryptographic file access control, login agents
- **Application segmentation and execution control** – isolating applications from each other, segmenting user access to only needed applications, preventing running of applications that have not been whitelisted
- **Operational and forensic analysis** – SIEM tools, external threat intelligence, vulnerability scanning, host and application monitoring, network taps, and forensic tools
- **Network encryption** – application layer encryption, VPNs, and other encryption tunnels
- **Trust and policy engine** – organizational risk analysis, principal vulnerability analysis, behavioral analysis

2.3 NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY (NIST) MODELS

NIST has released Special Publication SP-800-207 Zero Trust Architecture [7], as of this writing in its second draft. Within that document, they define three different variations on deployment of components for a ZT network. These models vary in the placement of the Policy Enforcement Point, from being very integrated with the resource and end-user device accessing said resource, or detached completely from these components, instead residing as a middle-man in the network. Since the goal of the NIST document is to be independent of commercial products, these models are deliberately generic, which can make it difficult to understand how they apply to the real-world implementations of ZT and existing products on the market. What these models do not show is the variation in the network stack where the enforcement may be done, and how ZT can be applied in places other than the network.

2.3.1 Device Agent/Gateway-Based Deployment

The first NIST model described is called the "Device Agent/Gateway-based Deployment" [7]. As shown in Figure 2, in this deployment model, policy enforcement is highly integrated with both the endpoint user device (marked "Enterprise System") and with the data resource or application the user is attempting to access. Enforcement is therefore distributed throughout the network resources and applications. This model allows for the maximum amount of control since the agent has contextual information on both the identity (user and enterprise system device) and the resource (application or data store) and can deny the access attempt at either end. However, this also requires the highest level of integration, since the end system must have an agent installed on it, and the application or data resource must interact or be fully integrated with the gateway. Google's ZT implementation, BeyondCorp, is mostly using this model at the application layer of the network. In BeyondCorp, Google has the ability to modify the browser on the user side and has modified their applications to work with BeyondCorp policy enforcement [4].

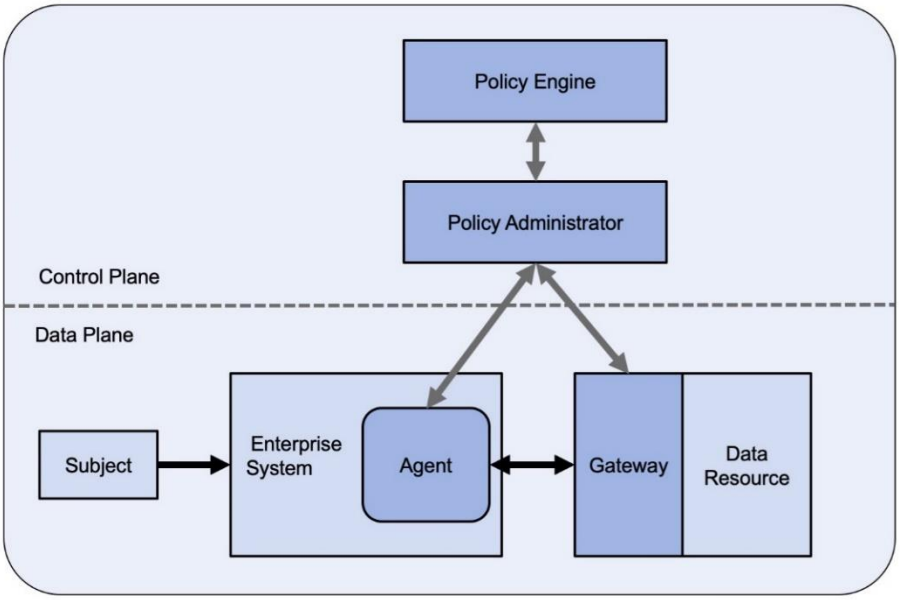

Figure 2. NIST "Device Agent/Gateway-Based Deployment" Model [7].

2.3.2 Enclave-Based Deployment

The second NIST deployment model is called "Enclave-based Deployment" [7]. As shown in Figure 3, this model still uses an agent on the user's device, but places the policy enforcement as a gateway in front of a resource or enclave of resources. This means that unlike the Device Agent/Gateway-based Deployment model, this model does not require a tight integration with the resource or application. However, as a result, this creates a zone of implicit trust between the gateway and the resource, and does not allow for as fine-grained policy decisions as the first model, since it does not have the additional context of the application/resource (e.g., what file the user is trying to access).

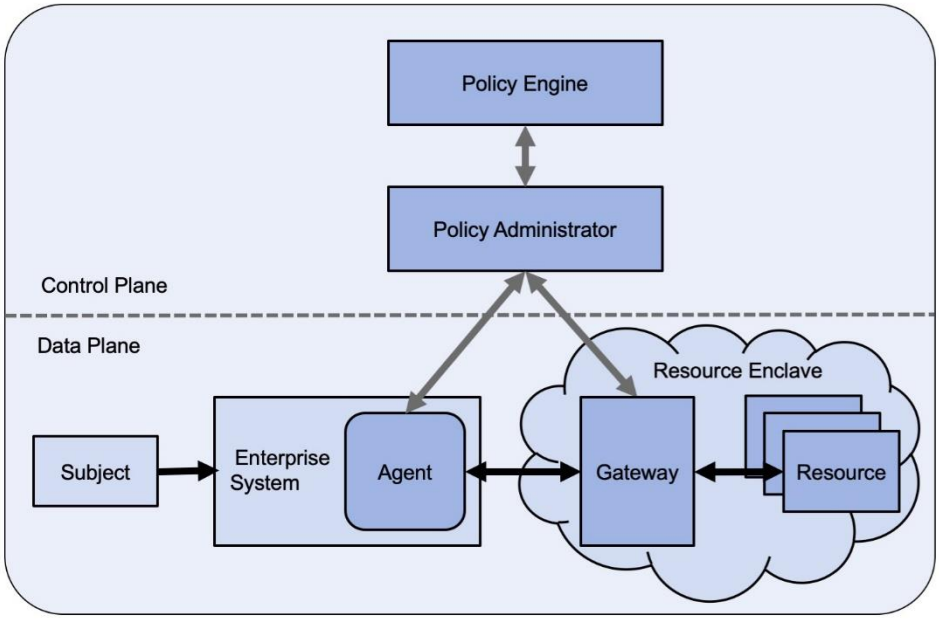

Figure 3. NIST "Enclave-Based Deployment" Model [7].

2.3.3 Resource Portal Model

The third NIST model is called the "Resource Portal-Based Deployment," as shown in Figure 4. Instead, in this case, the policy enforcement is not integrated with either client device or application/resource. Instead, a gateway that can control access to this resource is placed in the network path. This model does not require the application to be modified, or the user's device system to run any special software. However, it also provides the least amount of fine-grained control over access to the resource, and little to no contextual information from either end that can be used to make intelligent trust decisions. This model is the one first proposed by Forrester in their original concept, where resources were isolated on Very Large Area Networks (VLANs) and a segmentation engine, most likely a Next-Generation Firewall (NGFW), in the middle determining which connections would be allowed.

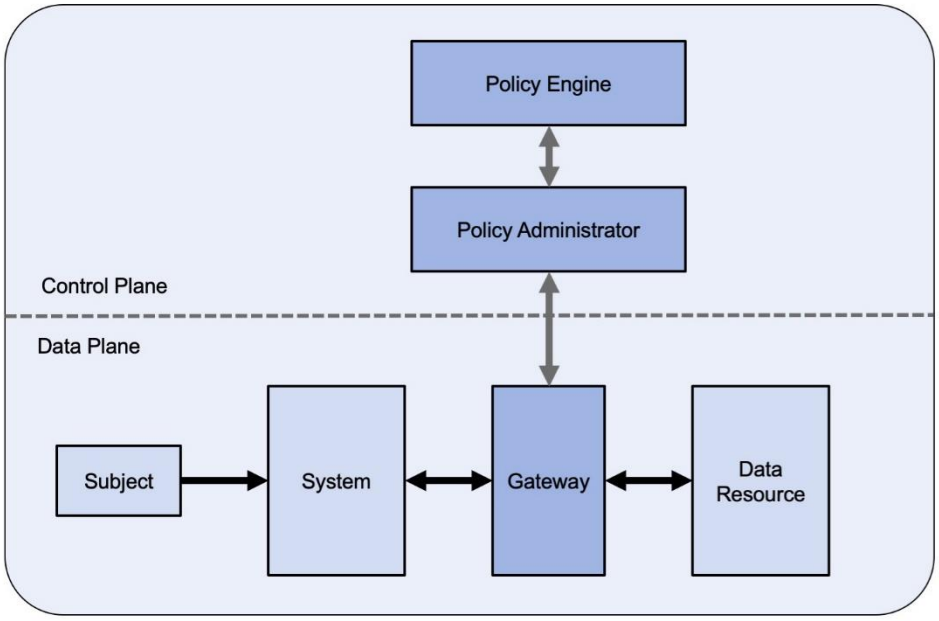

Figure 4. NIST "Resource Portal-Based Deployment" Model [7].

2.4 ARCHITECTURE CASE STUDIES

In this section, the four main ZT architectures in commercial deployment are described. Unlike the notional architectures from NIST described in the previous section, these are existing, real-world deployments.

2.4.1 BeyondCorp

BeyondCorp is Google's implementation of ZT following the Aurora attack in 2009, in which attackers gained the ability to alter Google source code [9]. Google wanted to eliminate the automatic assumption of trust given to users and devices simply because they were attached to the internal corporate network. BeyondCorp has three core principles [10]:

- A particular network connection must not determine which services a user can access.
- Access to services is granted based on what we know about a user and the device.
- All access to services must be authenticated, authorized, and encrypted.

BeyondCorp uses the Device Agent/Gateway-based Deployment NIST model. Figure 5 shows the components and access flow of BeyondCorp. While devices on the internal unprivileged network authenticate via 802.1x through a RADIUS server in order to access that network, all users have the same flow through the single sign-on server for authentication to resources and their access is explicitly allowed based on policy, rather than implicitly allowed simply because they are on a particular network. BeyondCorp relied heavily on the fact that Google applications and products are mostly internally

developed, already had their own single sign-on system, and thus could be modified to work with the new system. In addition, Google is authenticating the user at the application layer of the network, which they can rely on because their applications and resources are web-based. An enterprise that relies on other protocols will need a different model. Google has released some of their BeyondCorp technology as BeyondProd, a cloud-native security solution [11]. Google Cloud is FedRAMP High authorized, though it is unclear whether the BeyondProd offerings are included in this approval [12].

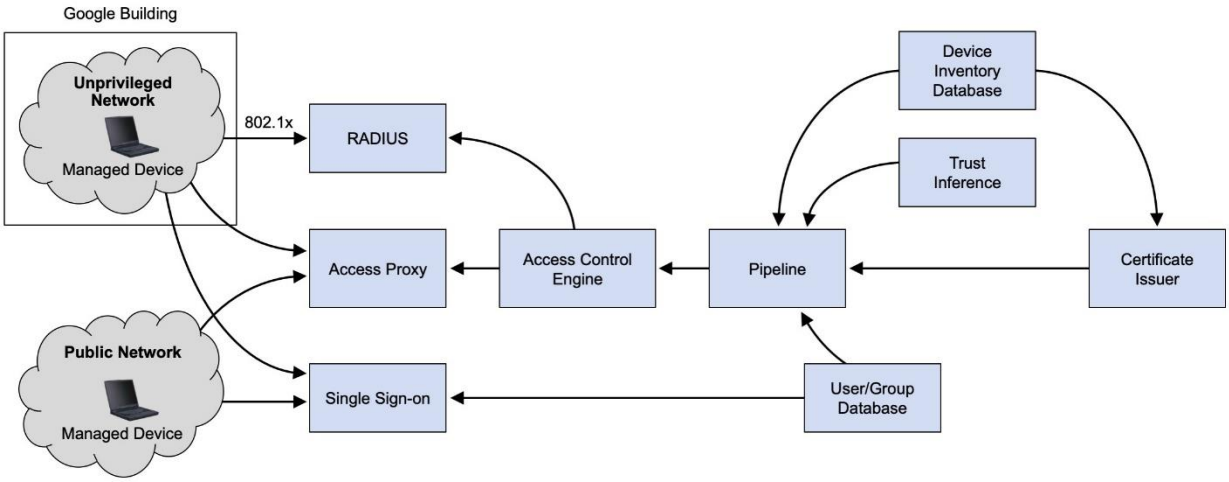

Figure 5. BeyondCorp components and access flow. (Figure from [14]).

For federal networks, the BeyondCorp model might be one that works well for a federal department or agency that is heavily public-facing with many cloud-based applications accessed by the public. In the current state of offerings, it cannot be run locally, it is only hosted by Google on their cloud infrastructure, and no other similar offerings are available from other companies. Utilizing the hosted version involves moving user-access-policy management controls outside of the organization to be physically hosted by the cloud provider, which for most organizations is not an acceptable solution.

2.4.2 NGFW/Forrester

Forrester Research originated the concept ZT in 2010 [1]. They developed a network model based on a centralized segmentation engine that separates the enterprise network into multiple "microcore and perimeter" (MCAP) segments, as shown in Figure 6. This allows the NGFW to enforce traffic rules between the MCAPs. This architecture conforms to the "Resource Portal" NIST deployment model. The advantage of this model is that it can be deployed with very few changes to the applications or resources the enterprise network is supporting. However, because the enforcement is being done in the network stack, this model can only enforce trust using the protocol information available in the data packets, which is a less granular level than architectures that integrate more closely with the endpoints and applications. The user is not authenticating with the segmentation engine; therefore, the segmentation engine cannot enforce policy based on user or device attributes. Policy is enforced using IP address as identity, unless the segmentation engine is integrated with other authentication technologies. Further integration between the firewall and other technologies, such as the VPN, Identity and Access Management (IDAM), or device management systems can help alleviate this problem.

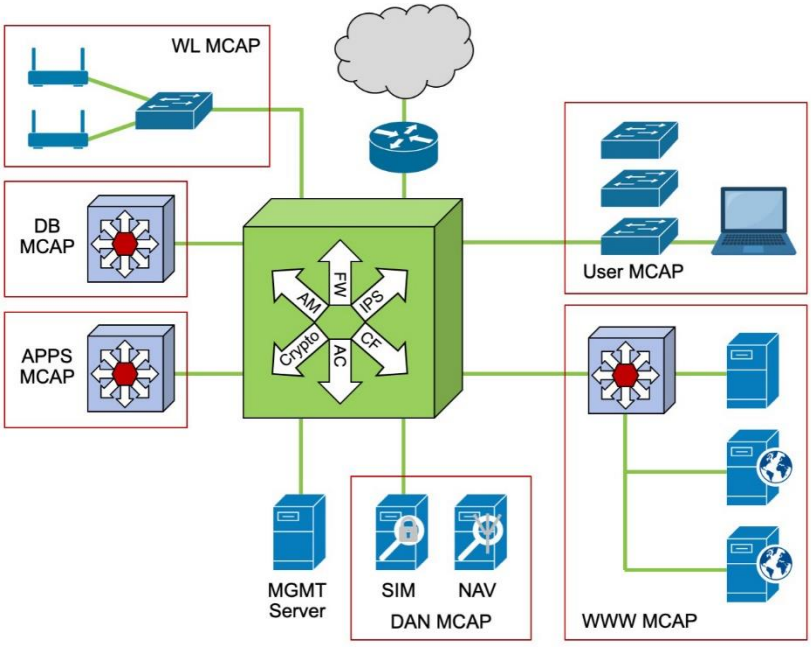

Figure 6. The Forrester/NGFW model uses a next-generation firewall at the center of the network as a segmentation engine, forming multiple MCAPs. (Figure from [1]).

For federal networks, this architecture is likely the lowest cost solution to deploy since it replaces/updates hardware in the center of the network. Many organizations already have enclaving architectures that fit well into this ZT architecture. This architecture would also be the easiest to deploy for a Bring Your Own Device (BYOD) or Internet of Things (IoT) solution by segmenting these devices into their own enclave/MCAP. However, it would not be the highest recommended solution as the level of granular control is not as fine as the other architectures and would require further integration with other technologies to achieve the equivalent levels of security as the other architectures.

2.4.3 Software-Defined Perimeter (SDP)

In 2013, the Cloud Security Alliance defined a new concept called Software-Defined Perimeter (SDP) [13]. Since then, several SDP solutions have come to market and it is now a viable solution for larger enterprises. SDP implementation can be considered a NIST Enclave-based deployment model, since it uses an agent on the endpoint device and an agent on the application side that is not integrated with the application, and therefore could be considered to fill the role of gateway. In some ways similar to the Forrester model described in the previous section, SDP operates as a central firewall segmenting the network, but is not integrated directly with the resource or application being accessed. The difference is that SDP creates an overlay network on top of the existing network infrastructure. The user authenticates to the SDP server, allowing the server to verify the user's identity and create a VPN tunnel between the user and the resource or application being accessed. This process is illustrated in Figure 7, which shows the SDP controller negotiating a connection between two hosts, ultimately resulting in a direct VPN connection between the two hosts.

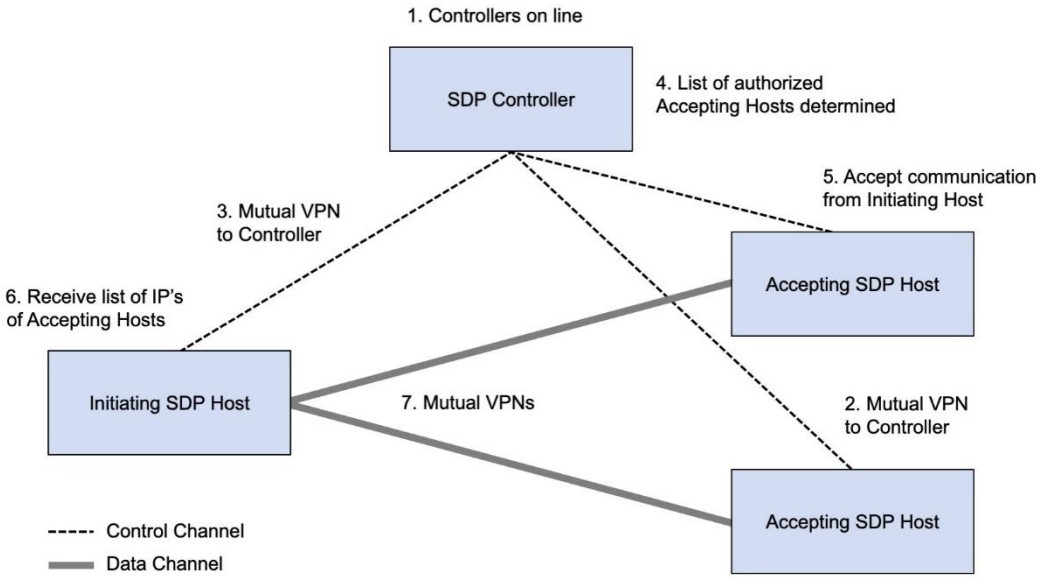

Figure 7. Workflow of the architecture of SDP. Figure from [13].

Unlike a traditional VPN, SDP does not allow authenticated users access to anything in the network by default, but rather enforces access based on attributes, roles, and/or user trust. The advantages of SDP are that it does not require integration with the application or resource, though it does require installation and configuration on both the resource server and on the user's device. SDP is a somewhat new concept and the technologies have not been on the market long, but have reached a point of being able to support enterprise systems.

For federal networks, this architecture is a good general choice. There are situations where others may be better, but SDP offers the best overall solution that will work with a variety of situations, doesn't require costly integration with applications, hosting on external cloud providers, or significant changes to users' working environment. This architecture can also be adapted for an organization with an extensive amount of IoT and other sensors by utilizing a gateway to act on behalf of those devices with the SDP system.

2.4.4 VMWare/NSX

Another real-world ZT architecture is VMWare's NSX deployment. This model, which would fall under the NIST Device-agent/gateway deployment model, is designed for enterprises utilizing a virtual desktop infrastructure (VDI) and predominantly virtualized systems. In this case, all resources and applications are hosted on virtual servers. Users authenticate to the VDI server and remotely access their virtual desktop. Running on each virtual desktop and each virtual server is VMWare's network and security virtualization (NSX) software. This software acts as a firewall, enforcing trust and policy decisions at all points in the network. A ZT Architecture using NSX is shown in Figure 8. This allows administrators to segment their network in an extremely fine-grained way, which is often referred to as micro-segmentation. An additional benefit of this approach is that the virtual desktops are uniform and controlled by administrators and can be refreshed or rebuilt on a regular basis, such as nightly. This prevents an adversary

who does compromise the system via the user's desktop (a very common point of entry via phishing, etc.) from gaining a persistent foothold in the network.

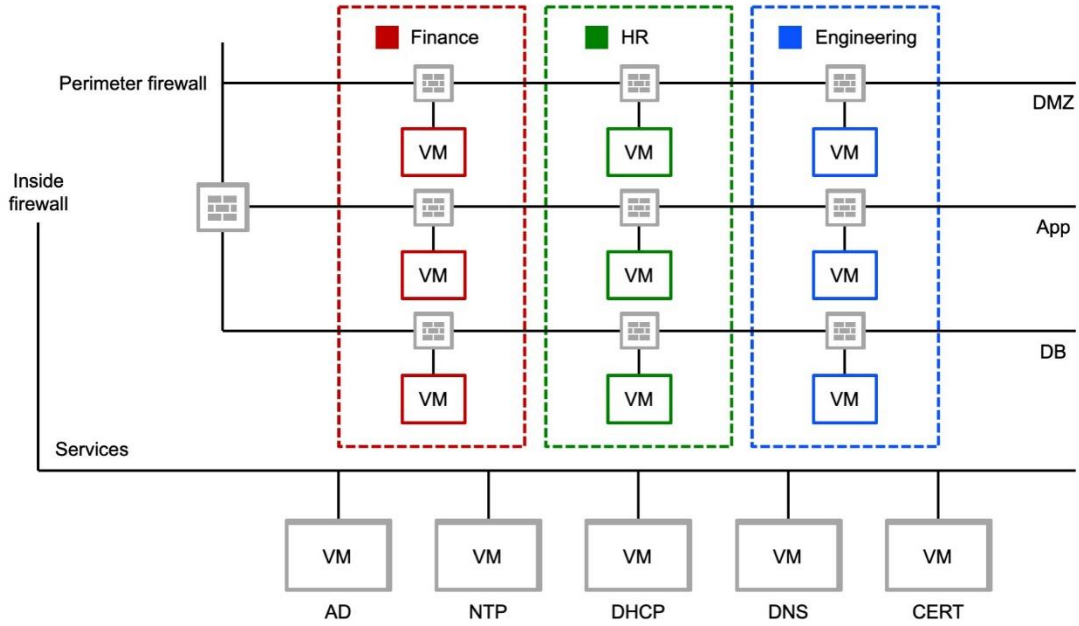

Figure 8. ZT model using NSX. (Figure from [15]).

For federal networks, this architecture is the best choice where heavy virtualization is already deployed or planned or for smaller organizations whose workforce has continual internet access. The virtual ZT architecture provides the most control and granular segmentation of all the architectures, but switching an organization over to a VDI-based architecture is a costly proposition if this organization is not already using it. Additionally, regardless of cost it would be a poor choice for an organization with extensive IoT devices or other sensors that cannot be virtualized. A larger organization could utilize VDI as part of a federated architecture.

2.5 FEDERATED ARCHITECTURES

Realistically in a large enterprise, it will be difficult to deploy one single ZT architecture enterprise-wide. Even if the intention is to move to a single architectural approach eventually, it is likely there will be a transition period. Therefore, a hybrid architecture, where several of the different models described above are federated together, will likely be what most organizations will want and need to do. However, most ZT technologies on the market today do not provide the ability to integrate with each other, or offer any guidance to customers on how to do so. As a result, there is no publicly documented real-world use case to guide this approach, and an iterated deployment with regular checkpoints and post mortems is recommended. ZT architectures are designed to be able to be deployed in stages, which allows for some amount of coordination between the ZT solutions and existing solutions, but this is different from federating across policy and trust decisions.

For federal networks, especially those that have many large sub-organizations with wide-ranging mission types, implementing a federated architecture may be the only reasonable choice. An agile approach that chunks stages and portions modestly and tractably, and uses all the retrospective features to learn from each iteration and release, is likely the best approach.

This page intentionally left blank.

3. IMPLEMENTING ZT ON FEDERAL NETWORKS

3.1 CHALLENGES

3.1.1 Wide Range of Choices with Little Organization

ZT is a concept, not a specific technology. However, there is no comprehensive ZT solution on the market. As a result of this, there is a wide range of choices as to how it is implemented. The diversity of the market creates a big challenge because very different technologies are labeled as "Zero Trust," yet cover a variety of different parts of what makes up a ZT architecture. Therefore, government agencies have the challenge of piecing together a full ZT implementation from a variety of products with often hard-to-determine specifications and sometimes conflated features.

3.1.2 Lack of Integration Caused by Vendor Lock-In

Despite the fact that there are many disparate solutions that cover different parts of the whole of ZT, vendors are determined to keep customers locked-in to their platform. This makes it very difficult to integrate solutions that may be necessary for a variety of different reasons. In government architectures especially, agencies may be working with what has already been purchased in the past or for them via purchasing programs. Additionally, government networks tend to be diverse, with a variety of technologies being used and therefore may need a diverse set of technologies to provide ZT.

3.1.3 Measuring ZT

While there is much discussion about ZT, suggestions for how to implement it, and technologies claiming to provide it, there is not currently a standard or metric allowing an organization to measure whether it has achieved the goal of "Zero Trust." Without a metric to measure success or at least a checklist of items that must be done, an organization is left attempting to fulfill a goal it cannot validate. Many concepts in ZT are not either on or off, but rather gradations. For example, how much is enough in terms of access segmentation? Is dividing the network into a few segments enough or should it be segmented into tiny pieces? The answer is likely not the same for every organization, as it will depend on what is being protected, the organization's risk profile, and the cost of implementing more advanced levels of segmentation.

3.1.4 Organizational Priorities

ZT technologies and policies come with a cost, both to the organization financially if new technology is being installed or existing technology reconfigured, and also in terms of extra effort and hassle to both administrators and users, e.g., two-factor authentication adding additional effort to every login. If the organization does not prioritize security and show this in their policies and decision-making, shortcuts will be taken that undermine the point of the ZT architecture to begin with. When security is made a top priority, IT administration can feel empowered to make choices that will improve security without fearing they will get backlash for doing so.

In addition, ZT principles come with a cost to user and especially admin productivity. Increased requirements for authentication may require more frequent inputting of credentials. Segmenting of access may mean having to request access or privileges more frequently than before. For administrators, the segmentation of privileges means having to go through extra steps, possibly log into different accounts or take extra measures to grant an account the necessary privileges to perform work. Finding ways to reduce or compensate for these disruptions is critical to productivity and to user compliance and satisfaction, but the reduction cannot come at the cost of the security itself, and that can be a challenge.

3.1.5 Lack of Independent Analysis

Most information about ZT comes from the vendors and companies that make ZT solutions themselves, which tend to be optimistic about their products. What is needed is an independent analysis of ZT architectures, done by someone without a financial stake in the outcome, in order to verify the claims that have been made about their security properties. Since ZT is a new concept, this verification needs to be done at a higher level than product verification—the concepts themselves that ZT is based on must be verified. We believe these are good practices, but at this point, they have not yet been proven to be so.

3.1.6 Scale of Monitoring Data

One of the core ZT principles is Continuous Monitoring and Adjustment. On its face, it seems like a simple best practice, but that level of monitoring creates a challenge in itself on how to deal with the sheer scale of data comprehensive monitoring requires. The Department of Homeland Security (DHS) Continuous Diagnostics and Monitoring (CDM) program is currently facing this challenge as it attempts to track data on a massive scale for every federal department and agency. Monitoring does not provide much benefit, however, if it cannot be used to adjust policies and access appropriately. So not only does the data need to be stored, it needs to be analyzed and available to the PEP in order to make valid decisions about which entities should be able to access which resources. Additionally, guidelines do not exist for exactly how much data needs to be monitored to comply with the tenets of ZT, and what data should be acted upon or considered for policy enforcement.

As the amount of data scales, there is a corresponding challenge in the analysis of that data. In order for a ZT architecture to utilize the captured data and make access decisions based on trust calculations, the analysis of the monitoring data must be automated and able to handle the large amount of data. Current capabilities for analysis of this type of monitoring data do not scale well and present a challenge.

3.1.7 Policy Conflict

Government organizations have very unique missions, and as a result may have policies that conflict directly with the principles of ZT. An intelligence agency may, for example, need to receive data from sources that cannot be authenticated. There are policy mandates that are outside the control of the organization, so that even if the organization wishes to modify policy to be in line with ZT, this could be a very lengthy and difficult process.

3.2 ARCHITECTURAL DESIGN CHOICES

This section will discuss the factors that go into choosing a ZT architecture.

3.2.1 Single vs. Federated

A ZT architecture can either be a single approach across the entire organization, or could be a federation of several approaches. Certain existing network architectures lend themselves better to particular ZT approaches, but often large organizations have diverse network architectures and may find that implementing an approach that federates several different ZT solutions may make more sense than trying to migrate the entire organization to one approach. However, with the current state of vendor lock-in in ZT solutions, federation carries increased challenges as well.

Factors to consider include:

- **Diversity of existing network** – Is the existing enterprise unified on a single network architecture, or are there sub-organizations with their own diverse solutions? Does the organization rely on the same authentication scheme?

- **Existing modernization/upgrade efforts** – Could a move to ZT dovetail with existing upgrade and modernization efforts? If the organization is already planning to unify its enterprise network, implementing a single ZT architecture could be part of these efforts.

- **Diversity of mission** – Does the organization as a whole have a unified single mission, or is it comprised of many very diverse mission types?

3.2.2 Starting from Scratch vs. Incremental Approach

In almost all cases, an incremental approach is necessary. Unless the enterprise architecture is being built from scratch, it is unlikely that it will be feasible to roll out a ZT approach from scratch. ZT is not a new capability being deployed, but an improved configuration of capabilities, many of which already exist and are deployed in a current enterprise network. Even if the architecture chosen is very different, users and resources will have to be incrementally moved from the old system to the new.

If it does happen that the network and enterprise architecture is being built from scratch, ZT can be built-in more easily from the beginning. ZT can be part of a larger effort to modernize or replace existing equipment. Any ZT architecture will benefit from being able to be built from scratch, but most particularly those that fall under NIST Model #1, Device Agent/Gateway-Based Deployment. In the Device Agent/Gateway-Based Deployment model, the policy enforcement points are tightly coupled with the user device endpoints at one end and the application/resource endpoints at the other end. This tight coupling is more work to do in an established system, and therefore will benefit the most from deploying from scratch. Of the architecture case studies discussed in Section 2.4, the VMWare/NSX architecture is one that also provides other benefits besides the ZT security. Moving to a VDI architecture allows for easier upgrades, for example, and many government organizations are considering or have moved to VDI for other reasons. The types of organizations that would benefit from this architecture are discussed in Section 3.4.

Since starting from scratch is a special case, most likely existing infrastructure will drive ZT architecture choices. For federal architectures, an incremental approach makes the most sense, allowing the organization to incorporate ZT into the existing phased rollout of capabilities under the Continuous Diagnostics and Monitoring (CDM) program.

3.2.3 What Applications are Supported?

Another factor to consider is which applications will be supported in the ZT architecture. Architectures that require tight integration with the application are more feasible if the applications being supported either already support the ZT technologies being considered or are under the organization's control to customize and modify code to integrate with the ZT technology. In government systems, this is less likely to be the case, even custom applications that may be under the organization's control do not necessarily have the development support available to make these changes.

If the applications being supported are Commercial Off-The-Shelf (COTS) products that the organization does not have the ability to modify and update, then a solution that is less integrated with the application and contains either a proxy element or is operating at the network level would make more sense.

3.2.4 What Devices will be Supported?

The devices that are to be used on the ZT architecture is a factor to be considered as well. If the organization has extensive VDI infrastructure and user devices are virtual, this means the organization has significantly more control of the applications installed on the device, and can easily refresh devices and segment access. If the organization has many laptops running the applications natively, capabilities to control and authenticate software running on these devices is more critical. Considering where devices will be used lets the organization prioritize technologies. If devices will connect from remote networks, those technologies are a priority. If all the devices will be used on internal networks, other technologies can be prioritized first. If users are going to be allowed to BYOD then how these technologies are going to be authenticated and secured must be a priority.

Internet of Things (IoT) devices by their nature have very minimal power and resources, and lack the ability to modify and customize them. This means that things like device authentication and end-to-end encryption can be much harder to implement and they cannot be treated the same as other types of devices. An organization with significant IoT assets will probably need to include technology designed specifically to handle these devices in their ZT architecture.

Management of Non-Person Entities (NPE), which encompass a wide range of devices, servers, and IoT devices is a big struggle for federal networks. Not all devices are tied directly with a user identity, and therefore, cannot utilize the PIV card that is required for authentication. Any ZT architecture should account for how NPE will be authenticated as well as how trust and policy will be affected by them.

3.3 IMPLEMENTING A ZERO TRUST (ZT) ARCHITECTURE

There are many options on how to go about implementing a ZT architecture, often designed to follow a specific technology or set of technologies. There is not a one-size-fits-all approach because organizations differ greatly in their mission, their enterprise systems, and their culture.

3.3.1 Planning Approaches

Taking a step back from the architecture itself, there are different ways to approach the problem. An organization can take the following different approaches:

- **Master Plan Approach –** an organization takes the time to evaluate their own current network and security policies, attack surface and budget. They investigate available ZT architectures and technologies and build a master plan of what they want their ZT architecture to be, then deploy it.
    - *Advantages*: Complete and thorough process will take many factors into account. The resulting plan is likely to be the best available for the organization.
    - *Disadvantages*: Determining a grand best plan is slow and costly, especially if underfunded. A team of people will need to be dedicated to this, as well as having support throughout the organization. There is a risk that the plan will be out of date and the chosen technology no longer state-of-the-art by the time it is actually implemented.
- **Use Case Approach –** instead of attempting to encompass the entirety of the organization, instead focusing on building use cases or Agile-style user stories, prioritizing these cases and implementing solutions for them one at a time.
    - *Advantages:* Sees much quicker results by focusing on individual use cases and the implemented solutions are more likely to provide specific results. This can be implemented with a small team, or by having different parts of the organization handle their own relevant use cases. It provides a way to make progress on a small budget.
    - *Disadvantages:* The disadvantages are that without a larger plan the organization runs the risk of implementing disjoint or redundant technologies. Coordination can be difficult if not centralized team is in charge. Use cases can be too narrow and not encompass enough of what needs to be done to deploy a complete system.
- **Principle-Focus Approach –** working through the various principles of ZT one at a time, focusing on improving the organization's security for that principle, then moving on to the next. For example an organization could begin by focusing on Universal Authentication, implementing multi-factor authentication, single sign-on, device authentication, and focus on all resources requiring this authentication for access. When that is in place, the organization could next focus on segmenting access and so on.
    - *Advantages*: Sub-divides the problem to allow for a more focused effort, allows organization to apply the principles more uniformly across the entire organization.

- *Disadvantages*: Some principles are tied together and not everything can be implemented incrementally that way. May require circling back to previous principles to implement things not available until other principles were implemented.

- **Domain-Focus Approach –** picking a domain to focus on first, deploying ZT technology across the organization for that domain. The domain could be a number of different factors, such as location, focusing on implementing ZT for a particular site or building; or it could be for groups within the organization, division by division; or technological domain: focus on the servers and desktops first, then on mobile devices, then cloud services.

  - *Advantages:* Sub-divides the problem for a more focused effort, which can be done with a smaller budget. Dividing by technology allows for a focus on a smaller number of solution products. Dividing by location allows teams to be physically co-located and more efficient.

  - *Disadvantages:* Choices for some domains may impact others, not everything may be taken into account when a technology is chosen.

3.3.2 Steps for Implementation

This section describes a set of steps for implementing a ZT architecture. Since organizations vary widely, it will likely need to be adjusted accordingly, but this is a starting point.

Step 1 – Choose a Planning Approach

Section 3.3.1 describes a number of different planning approaches to determine which would make the most sense for your organization. Having a clear plan for how you will proceed will help keep the project on track.

Step 2 – Inventory

Regardless of which planning approach is chosen in Step 1, it is critical to determine what the scope of the ZT architecture will entail and how the various parts will be prioritized. This step includes two parts: gathering information about what your organization has and deciding how it will be prioritized for your chosen approach. This needs to be an organization-wide effort to ensure you have not missed critical resources that part of the organization is relying on. The information technology (IT) function will likely have most of the answers, but since implementing a ZT architecture affects users as well, it is important to verify with all stakeholders that nothing important has been left out. The following questions help to inventory what technology and domains your ZT architecture must account for:

1. What applications and resources does the organization have?

2. What different types of endpoints and servers does the organization support?

3. What classes of users utilize these applications and endpoints?

4. What types of authentication are in use?

5. What types of networks are we currently using?

6. What types of data need to be protected?

Step 3 – Divide Scope for Approach and Prioritize

Having chosen an approach and inventoried what assets must be included, this inventory should be divided up based on your approach. Even if a master plan approach is chosen, dividing up the workload by ZT principle or a domain is recommended to work through the planning stage.

For example: If the approach chosen was principle-based, then for each of the categories described in Step 2, the inventory should be prioritized, so that while the organization is going to focus on the principle of Universal Authentication first, the order of how that will be applied will be to the most critical combination of applications, users, and devices first.

Step 4 – Create Use Cases

Even if you have not chosen to take a use-case centered approach, creating use cases is a valuable tool for thinking through how ZT will be implemented and will affect your users. If you are taking a use case based approach, this should be done for the entire scope of effort. If you are using a master plan, domain or principle based approach this could be done as part of the iterative Step 5.

Step 5 – Iterate over each Focus Area

Iterate over the following steps for each focus area in your chosen approach. If you are taking a master plan approach, following Steps 5a-d in your planning phase to create the master plan, and then Steps 5e-f in your implementation.

Step 5a – Set Success Criteria

This is a critical step, but one that has the least available resources to assist, as described in the challenges section and the subject of ongoing research. Even if working with imperfect knowledge of what constitutes a successful ZT architecture, setting some basic ideas for what success means will help guide the organization in its implementation.

Step 5b – Create an Implementation Plan

For the chosen focus area, identify how each ZT principle can be applied to that focus area. Or if taking a principle-based approach, how that principle can be applied to each of the different items in your organizational inventory.

Step 5c – Identify Existing Valid Use and Determine Policy

Identify how the existing focus area is being used, what uses are valid, and which are unnecessary. Design a policy that encompasses the valid use cases.

Step 5d – Get Stakeholder Buy-in and User Awareness

This is a critical step, attempting to implement a ZT technology that will disrupt the mission will not go well if the stakeholders don't buy into why it needs to be done. Determine what training and awareness users need to have.

Step 5e – Deploy Technology in Audit-Mode

Deploy the technology or security control in a mode where the users have switched over to using it, but do not turn on enforcement at this time. This may not only apply to policy, but also to changing over to a new technology. Try to get users moved over while still having the old method to rely on. Enable warnings when using old technology that it will be deprecated soon and/or that the user is violating policy whenever possible.

Step 5f – Turn on Enforcement

When the team is sure that all functions are using the new technology/policy/method without issue, then policy enforcement should be turned on, or the old technology turned off.

3.4 EXEMPLAR FEDERAL AGENCIES

This section will discuss three different generic federal agencies as exemplars for illustrating the types of choices that make the most sense for organizations with these missions and needs. While they are referred to as agencies, these could stand in for sub-agency units as well, such as a component, division, or directorate. The example agencies, components, or directorates mentioned under each exemplar are merely to indicate the types of organizations that may fit this exemplar; no analysis of the organization in question has been done.

3.4.1 Public Service Agency Alpha

Alpha agency is one whose mission is mainly focused on interacting directly with the public to provide a public service. Examples of agencies with a mission similar to Agency Alpha might include agencies like the Internal Revenue Service (IRS), Federal Election Commission (FEC), and DHS Components or Directorates like United States Citizenship and Immigration Services (USCIS) or Office of Partnership and Engagement. This exemplar public service-focused agency has the following core missions that relate to ZT:

- Disseminate information to the public
- Collect information from the public
- Protect personally identifiable information

This exemplar public service-focused agency has the following organizational properties:

- Workforce mainly distributed at agency offices nationwide
- Similar in many ways to typical corporations

ZT Recommendations: Any of the NIST models or architectures described in the case studies in Section 2 could be used by Agency Alpha. Particularly because Agency Alpha's workforce is generally continuously connected to the internet, a virtualized solution like the VMWare NSX architecture described in Section 2.4.4 would provide a significant amount of control over the flow of information Agency Alpha is charged with protecting. If Agency Alpha has applications where it provides certain application access to members of the public, a BeyondCorp-style solution as described in Section 2.4.1 would be very applicable, as it provides easy external access to applications and tight control over access to them. At this time, Google has brought that technology to market as their BeyondProd offering, but it cannot be run locally and requires moving user-access-policy management controls outside of the organization, which for most government organizations is not an acceptable solution. In the future, this technology may be offered in a way that meets federal government requirements, at which point it could be a strong choice for an Agency Alpha. An SDP-based or Forrester/NGFW solution would be good choices for organizations that already utilize a lot of enclaving in their existing architectures.

3.4.2 Public Safety with Field Agents Agency Beta

Agency Beta is one whose mission is mainly focused on protecting the public by operations that are dispersed geographically and that utilize a lot of field agents. Examples of agencies with a mission similar to Agency Beta might include agencies like the Federal Bureau of Investigation (FBI), and DHS Components like Transportation Security Administration (TSA) and Customs and Border Protection (CBP). This exemplar public safety-focused agency has the following core missions that relate to ZT:

- Collect information from and coordinate the actions of field agents
- Protect personally identifiable information and sensitive information
- Collect sensor data and analyze for potential threats

This exemplar public safety-focused agency has the following organizational properties:

- Workforce mainly distributed at field sites nation-wide or world-wide
- Workforce is only intermittently connected to the internet
- Rely on sensor networks of unique types of sensors
- Extensively utilize IoT technology

ZT Recommendations: In the case of Agency Beta, a ZT architecture like the VMWare NSX architecture described in Section 2.4.4 is not recommended, as virtualization requires constant connectivity, which field agents will not have. In addition, IoT and sensor networks may not be virtualizable. Additionally, a Forrester/NGFW solution, as described in Section 2.4.2, would be possible by segmenting the IoT devices on their own networks, but harder to secure IoT devices from traffic from one another within the segmented enclave. The recommended solution for an organization like Agency Beta is one that is SDP based, as described in Section 2.4.3, because this provides the most flexibility for incorporating IoT devices and sensors that cannot be modified or have software agents installed directly on them. The IoT device is enclaved in the network by a gateway that creates the encrypted tunnels using SDP on behalf of the device.

3.4.3 Larger Umbrella Organization Agency Delta

Agency Delta is an exemplar of a larger organization with mixed-mission sub-organizations. This agency has a significant amount of workforce operating in a typical enterprise environment within the umbrella organization, but also has diverse sub-organizations with missions that look more like Agency Alpha and Beta. Examples of agencies with a mission similar to Agency Delta might include the Veterans Administration, Department of Agriculture, and DHS itself with its headquarters and components. This exemplar umbrella agency has the following core missions that relate to ZT:

- Agency Alpha's mission set
- Agency Beta's mission set
- Coordination between sub-organizations
- Compartmentalization of information between sub-organizations

This exemplar public safety-focused agency has the following organizational properties:

- Workforce with disparate situations (field work and office work)
- Extensive enterprise-style networks and applications
- Variety of sensor networks with different missions
- Reliance on IoT devices of vastly differing types

ZT Recommendations: Of the four ZT architectures presented in Section 2.4, only an SDP-based architecture could reasonably meet the needs of an organization with both Alpha and Beta style sub-organizations. However, given the nature of an organization of this type, attempting to unify competing interests, it may be extremely difficult to unify the entire organization under a single ZT solution. We recommend that an agency like Delta should consider developing a federated architecture bringing together two or three of the other architectures to serve the needs of sub-components. Without a publicly documented real-world use case to guide this approach, an agile, iterated deployment that chunks tractable stages and utilizes regular checkpoints and post mortems to learn from each iteration is recommended.

3.5 BEST PRACTICES

At this point there have not been many examples to utilize in determining best practices for implementing ZT Architectures. This section describes those we have come across thus far.

3.5.1 Security from the Top Down (Priority for Leadership)

Making security a priority is critical. If senior leadership is supportive of security measures, even if they inconvenience management themselves, the deployment is more likely to be successful. If senior leadership is constantly demanding exceptions be made, security will likewise be weakened. An example of this seen in action was one government organization that said that they conducted regular phishing exercises (where an external company sends phishing emails to evaluate who clicks on them and is likely to fall prey to a real phishing attempt). When users failed the phishing exercise, they were given supplemental training. If they continued to fail their access was severely limited. This was a great example of implementing trust as context-based access control – access is restricted if users do not meet the criteria of passing phishing exercises. While this was only a small number of users, those users were the ones making the organization the most vulnerable to that form of attack. In extreme cases, this meant the users could not do their jobs. Instead of demanding exceptions, management of this organization chose not to put the entire organization's security at risk because a handful of users were not able to understand or follow policy.

For the federal government this is even more important to do and challenging to implement because of the nature of senior leadership postings where personnel move to new positions after a relatively short tenure, when compared to industry.

3.5.2 Collaborative Red Teaming

A critical best practice is having an external red team analyze and attempt to break the security of an organization's enterprise. Government organizations are already required to do regular Federal Information Security Management Act (FISMA) audits. However, these interactions are evaluations, therefore the organization's motivation is to do well in this evaluation. While these evaluations are useful for external analysis of an organization's security posture, they are less useful for finding new flaws than a collaborative effort. In collaborative red teaming, the external red team sits down with the organization's security team and together they determine the potential ways the organization could be compromised, which the red team attempts. If the red team is unable to penetrate a level of security, the organization gives them a foothold slightly further into the network and they resume attempting to penetrate further. This is also sometimes called purple team, due to the mix of offensive red team finding vulnerabilities and defensive blue team recommendations. This allows testing of the defense-in-depth capabilities. Even if the "outer perimeter" or the first defenses an attacker would encounter are strong now, future flaws may be discovered, so testing of the next layer beyond is critical. This process works best if it is done regularly, such as every six months, to test that new measures put in place to address previous red team findings are working. For federal organizations, this collaborative red team can be hired by an industry consulting company or the organization can take advantage of the National Cybersecurity Assessments and Technical Services (NCATS) program offered by DHS.

3.5.3 Full Utilization of Existing Resources

Many organizations, especially those in the government, may already have resources at their disposal that could be utilized to implement a ZT Architecture. Cataloging what existing resources are available and investigating how what the organization already has that could be used to implement ZT will help keep the cost of ZT down. Most organizations do not fully utilize the features built in to operating systems and enterprise software they already have. Sometimes existing technology only needs a small amount of user interface (UI) built on top to make it more useable. For example, one organization utilized existing Active Directory functionality to implement a role-based permission scheme that allowed their administrators to grant themselves privileges temporarily for the work they needed to do. These permissions later expired, preventing attackers from leveraging that access. The underlying functionality was available already—they simply built a basic UI to make the experience easier. For federal networks, procuring products have another level of difficulty due to regulations and budgeting, so finding a way to use what is already available lets the organization make progress toward ZT despite delays.

3.5.4 Lock Down any Unnecessary Access

New ZT technology is not necessary to take basic steps to lock down any unnecessary access within the enterprise system. Mostly what newer technology does is make it easier to manage this process. However, configuring switches to drop all connections to server ports not in use or between IP addresses that have no need to communicate can easily be done with existing systems. Utilizing permission schemes in applications like SharePoint to compartmentalize access to data should be done. Essentially making the default access for networks, data, devices, and applications whitelist only. In government organizations, where heavyweight new initiatives can take a long time to get off the ground, finding ways to lock access down is a lightweight project that can be done quickly yet provide significant security benefit.

3.5.5 Automation

Much of ZT is about having very restrictive policies in place to keep access to data and applications to only those users who need it to get their work done. In practice, maintaining a restrictive system on a large scale is difficult; often this results in more lax policies in order to make maintenance of them less cumbersome. To prevent this, identifying manual processes for managing policies and automating them will give a ZT architecture a better chance at succeeding at securing the organization by preventing the kind of relaxation of security that happens over time due to heavy manual process requirements. For federal networks, this may mean having to train or hire contract personnel.

3.5.6 Secure Contingency Planning

Access Segmentation is a critical principle for ZT architectures, and it should not just be applied to users but to administrators as well. Segmenting elevated privileges so that no one account can access the entire system provides significant protection. However, it also can increase the risk to the organization if critical accounts get locked out for technical reasons or because the owner of that account is suddenly incapacitated or unavailable. It is therefore critical to plan for these potential catastrophes by implementing “break the glass” style measures to allow critical access when needed.

One way of implementing such a plan is to create a set of accounts, which together have access to every critical resource. These should be secured with long, random passwords that are then kept printed out in a physically secure, non-cyber-accessible location, like a safe, against potential future need. Since these accounts are only used in an extreme emergency, they should be flagged and monitored and any use should send up massive alarms, to avoid the accounts themselves being abused. If they are ever used, the passwords should immediately be changed, considered to be one-time-use only. Since federal organizations already generally have physical security for sensitive information with access control, this can be integrated into that system.

3.5.7 Incremental Rollout

ZT advocates very stringent security controls, especially around accessing the network and resources. This can be make rolling out a ZT architecture very disruptive, as crafting the policy to allow the minimal amount of access necessary is a detailed process that is highly error-prone. As a result of this, it is best practice to roll out any ZT architecture incrementally in terms of the applications/resources being migrated and how the policy enforcement is deployed. This incremental approach was originally pioneered by Google in their BeyondCorp rollout [14].

First, move incrementally in terms of which resources are accessed through the ZT architecture. An organization can prioritize this in different ways, either by the number of users affected, the value of the assets, or the ease/cost of integration with the ZT system. It is recommended at least the first few applications/resources chosen should have a small number of users and be less critical to get the process down. After that, the order is more organizational choice, but focusing on moving one application/resource at a time keeps the number of affected users lower and allows the ZT migration team to learn and adapt as it proceeds.

Second, while migrating users of a specific application or resource to the ZT architecture, there is an incremental process by which ZT is applied to that resource. In order to avoid disruption to users, it is best practice to deploy any Policy Enforcement Point first in an audit-only mode. The restrictive policies are put in place, but do not prevent access; instead they simply raise flags. The ZT migration team works with users and with the resource owners to eliminate any policy-violation flags, either by determining the policy needs to change to accommodate legitimate work, or that the access being attempted is not necessary and educating the user of this. Once regular work can proceed without any policy violations occurring, then policy enforcement is turned on.

For federal networks, this incremental rollout can allow for working with sub-organizations to bring them onto the ZT network gradually while working closely with the leadership and users of that sub-organization.

This page intentionally left blank.

4. CONCLUSION

This document reviewed what ZT is, outlining the five core principles of a ZT architecture: universal authentication, access segmentation, least privilege authorization, encryption everywhere, and monitoring everything. ZT does have some shortcomings, mainly in a lack of definition, unclear criteria, and lack of knowledge in the degree of benefit to an organization.

The dimensions that ZT architectures differ from one another were covered: the networking stack layer, location of policy enforcement, and enterprise system domain. The different NIST ZT deployment models were described, as well as several architectural case studies into BeyondCorp, NGFW/Forrester, SDP-based, and VMWare/NSX.

Many challenges exist in deploying a ZT architecture. The market has a lot of choices with little organization or clear understanding of what various technologies cover. Vendor lock-in is a serious problem, making it very difficult to fit technologies from different vendors together. Specifications do not exist to measure the level of ZT or inform an organization as to what still needs to be done to achieve the organization's security goals. The cost in terms of user and administrator time and effort can be high and therefore, any ZT effort must be in line with the organization's priorities to be successful. Lastly, there is a lack of any independent analysis into the claims made by the ZT concept, or those made by individual technologies specifically.

Choices that need to be evaluated when choosing an architecture were covered. Whether a single solution or a federated architecture would be better depends heavily on the diversity of the existing enterprise systems. If an organization happens to be starting from scratch, then ZT can be built in from the beginning, but otherwise it is much more likely that an incremental approach will need to be developed. The organization must take into account the types of applications and devices that need to be supported when choosing a ZT architecture, because some architectures require much more integration with the application and device than others. Approaches for implementing a ZT architecture were described, with some recommended steps.

Recommendations were made for three exemplar federal agencies. For a public service agency whose main mission is to interact with the public, any of the architectures would meet their needs, and which should be chosen depends heavily on their existing infrastructure. However, a virtualized architecture like the VMWare/NSX would provide the most benefit, though it may be the most difficult to migrate to. For a public safety-focused agency with many field agents who do not have continuous internet access and utilize many sensors and IoT devices, an SDP-based solution is recommended. For a larger umbrella organization, which may have sub-organizations as the previous two exemplars, a federated architecture is recommended to be able to accommodate the diversity of requirements.

Lastly, recommended best practices were described. Making security a priority from the top down is extremely important, as pressure from above to relax security restrictions undermines ZT efforts. Collaborative red teaming can help an organization verify its efforts are worthwhile and prioritize future technology deployment, as well as increase the knowledge level of its cyber security administrators. Organizations should make best use of the resources they already have at their disposal, as often the principles of ZT can be enacted using what is already in existence and a new solution does not need to be deployed. Locking down access to communications, software, devices, and data is the key piece that is both difficult to do and most beneficial in terms of security. It is recommended that manual processes for maintaining policy and access be automated.

GLOSSARY

Agent	A piece of software installed on an endpoint in order to collect information for management of that device.
Application	A piece of software either run directly on a user's device (such as a desktop or laptop), or run on a server and accessed remotely.
Authentication	The verification of the identity of an entity.
Authorization	The verification of the ability for an entity to access a particular resource.
Endpoint	A device that is at either end of a network connection.
Gateway	Software that serves as a "bump in the wire" between a server and user connections in order to act on that server's behalf for functionality the server doesn't natively support.
Identity	A generic term for an authenticated entity, such as a user, an automated account or a service that has to be authenticated to access resources.
NSX	Brand name of VMWare's Network Virtualization and Security Platform
Policy	A rule or set of rules that determine whether or not a particular identity may access a given resource.
Policy Enforcement Point	A point in the enterprise system where policies are enforced, e.g., a firewall allowing/dropping a network connection or a gateway allowing/refusing connection to a resource.
Resource	Any asset within an organization that is accessed in order to complete work. This could be an application, a database, a cloud service, or computing platform.
Workload	A unit of computing function or application that can be separated from others that may run on the same infrastructure.

This page intentionally left blank.

LIST OF ACRONYMS

BYOD	Bring Your Own Device
CBP	Customs and Border Protection
CDM	Continuous Diagnostics and Monitoring
COTS	Commercial Off-The-Shelf
DHS	Department of Homeland Security
FBI	Federal Bureau of Investigation
FEC	Federal Election Commission
FISMA	Federal Information Security Management Act
IDAM	Identity and Access Management
IoT	Internet of Things
IP	Internet Protocol
IPSec	Internet Protocol Security
IRS	Internal Revenue Service
IT	Information Technology
LAN	Local Area Network
MAC	Media Access Control
MCAP	MicroCore And Perimeter
NCATS	National Cybersecurity Assessments and Technical Services
NGFW	Next-Generation Firewall

NIST	National Institute of Standards and Technology
NSX	Brand name of VMWare's Network Virtualization and Security Platform
OPM	Office of Personnel Management
PEP	Policy Enforcement Point
SDP	Software Defined Perimeter
TCP	Transmission Control Protocol
TSA	Transportation Security Administration
UDP	User Datagram Protocol
UI	User Interface
URL	Uniform Resource Locator
USCIS	United States Citizenship and Immigration Services
VDI	Virtual Desktop Infrastructure
VLAN	Very Large Area Network
VPN	Virtual Private Network

REFERENCES

[1] J. Kindervag, "Build Security Into Your Network's DNA: The Zero Trust Network Architecture," Forrester Research, 2010.

[2] McAfee Labs & Mcafee Foundstone Professional Services, "Protecting Your Critical Assets: Lessons Learned from "Operation Aurora," 2010. [Online]. Available: https://www.wired.com/images_blogs/threatlevel/2010/03/operationaurora_wp_0310_fnl.pdf. [Accessed 03 02 2019].

[3] A. Ellis, "Dark Reading," 20 02 2019. [Online]. Available: https://www.darkreading.com/threat-intelligence/9-years-after-from-operation-aurora-to-zero-trust/a/d-id/1333901. [Accessed 03 02 2020].

[4] R. Ward and B. Beyer, "BeyondCorp: A New Approach to Enterprise Security," *;login:*, vol. 39, no. 6, pp. 6-11, 2014.

[5] C. Cunningham, "The Zero Trust eXtended (ZTX) Ecosystem," Forrester, 2018.

[6] U.S. House of Representatives Committee on Oversight and Government Reform, "The OPM Data Breach: How the Government Jeopardized Our National Security for More than a Generation," U.S. House of Reprentatives Committee on Oversight and Government Reform, Washington, DC, 2016.

[7] S. Rose, O. Borchert, S. Mitchell and S. Connelly, *Draft (2nd) NIST Special Publication 800-207: Zero Trust Architecture,* Washington, DC: NIST, 2020.

[8] Markets and Markets, "Zero-Trust Security Market by Solution Type, Deployment Type, Authentication Type, Organization Size, Vertical And Region - Global Forecast to 2024," Markets and Markets, 2019.

[9] K. Zetter, "'Google' Hackers Had Ability to Alter Source Code," 03 03 2018. [Online]. Available: https://www.wired.com/2010/03/source-code-hacks/. [Accessed 11 03 2020].

[10] T. Desikan and T. Desikan, "What the heck is Zero Trust Security? How is it related to BeyondCorp? When can I have it?," 04 06 2019. [Online]. Available: https://blog.banyansecurity.io/blog/beyondcorp-zero-trust. [Accessed 27 04 2020].

[11] Google, "BeyondProd: A new approach to cloud-native security," 20 04 2020. [Online]. Available: https://cloud.google.com/security/beyondprod. [Accessed 05 05 2020].

[12] Google, "Techwire - Google Cloud Platform is now FedRAMP High authorized," 04 12 2019. [Online]. Available: https://www.techwire.net/sponsored/google-cloud-platform-is-now-fedramp-high-authorized.html. [Accessed 05 05 2020].

[13] Cloud Security Alliance Working Group, "Software Defined Perimeter," Cloud Security Alliance, 2013.

[14] R. Ward and B. Beyer, "BeyondCorp: A New Approach to Enterprise Security," *;login:*, vol. 39, no. 6, pp. 6-11, 2014.

[15] K. Kumar, "Micro-segmentation of Applications using Application Rule Manager," 04 04 2017. [Online]. Available: https://blogs.vmware.com/networkvirtualization/2017/04/microsegmentation-arm.html/. [Accessed 28 04 2020].

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
30-7-2020	Technical Report	

4. TITLE AND SUBTITLE

Zero Trust (ZT) Concepts for Federal Government Architectures

5a. CONTRACT NUMBER
FA8702-15-D-0001

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

K.D. Uttecht

5d. PROJECT NUMBER
10267

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

MIT Lincoln Laboratory
244 Wood Street
Lexington, MA 02421-6426

8. PERFORMING ORGANIZATION REPORT NUMBER

TR-1253

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Department of Homeland Security
2707 Martin Luther King Jr Ave SE
Washington, DC 20528-0525

10. SPONSOR/MONITOR'S ACRONYM(S)
DHS

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release: distribution unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This report describes the concept of Zero Trust (ZT), based on the key idea that an organization should not implicitly trust any network traffic, device, or user solely based on their physical or logical network location. Instead, ZT focuses on protecting resources. It requires any and all communication to be between explicitly verified and authorized users and devices. Further, any and all communication should be monitored. ZT is often misrepresented as eliminating firewalls; it is more accurate to say ZT places firewall-like policy enforcement points throughout the network. This eliminates the traditional firewall as a gateway from outside to inside, but still provides the same filtering of traffic.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			**19b. TELEPHONE NUMBER** *(include area code)*
Unclassified	Unclassified	Unclassified	Same as report	58	

This page intentionally left blank.