INSTITUTE FOR DEFENSE ANALYSES

Zero Trust Technology Integration Issues

Kevin E. Foltz, Project Leader
William R. Simpson

August 2021

IDA NS D-22663

INSTITUTE FOR DEFENSE ANALYSES
4850 Mark Center Drive
Alexandria, Virginia 22311-1882

The Institute for Defense Analyses is a nonprofit corporation that operates three Federally Funded Research and Development Centers. Its mission is to answer the most challenging U.S. security and science policy questions with objective analysis, leveraging extraordinary scientific, technical, and analytic expertise.

About This Publication

This work was conducted by the IDA Systems and Analyses Center under contract HQ0034-14-D-0001, Project C5223, "ELS as a Zero Trust Architecture," for the IDA. The views, opinions, and findings should not be construed as representing the official position of either the Department of Defense or the sponsoring organization.

Acknowledgements

Kevin Garrison, Peter A. Kind

For More Information

Kevin E. Foltz, Project Leader
kfoltz@ida.org, 703-845-6625

Margaret E. Myers, Director, Information Technology and Systems Division
mmyers@ida.org, 703-578-2782

Rigorous Analysis | Trusted Expertise | Service to the Nation

Zero Trust Technology Integration Issues

William R. Simpson and Kevin E. Foltz

Executive Summary

Zero Trust (ZT) is a popular term within DoD these days. Many products market ZT as a key selling feature, but simply adding a ZT component to an existing non-ZT architecture does not make it ZT. ZT is a philosophy, an approach to security, and an integration of many security techniques. Based on more than a decade of designing, building, and testing a ZT approach for the U.S. Air Force, the sequence of short papers that follow dispel some of the myths and misconceptions that commonly arise with respect to ZT.

Why ZT? Why Now?

The first question about ZT simply asks "Why?" Why is it popular? Why is it popular now? Why should we use it? The short answer is that the current approach is simply not working, and ZT provides an approach better suited to current threats. Defining and protecting network boundaries relies on strong boundary protections. With mobile devices and work-from-home commonplace, the boundary is increasingly difficult to define or defend. In addition, attackers often take advantage of phishing to harvest credentials, which allows them to start their attack from the inside. As a result, we do not see them coming until they have already moved laterally on the internal networks and compromised multiple resources.

ZT is based on a premise that attackers have broken through defenses, are continuing to do so, and will continue to do so despite our best efforts to stop them. As a result, defense is not about defining secure spaces but is instead about every resource being its own secure space that requires security for each access. Security is done continuously, end-to-end, and bilaterally. The need for implicit trust is eliminated by establishing authorization at each resource request.

Although ZT is becoming popular now, it was needed years ago. The current approach was flawed from its inception, but those flaws, and the need for a new approach, are only becoming obvious now, as the number and sophistication of attacks increase.

Can Single Sign-On Be Used in a ZT Architecture?

Single sign-on (SSO) is a convenient way to centralize and standardize authentication for resources across an enterprise. A user need only authenticate once to access many resources. But this convenience comes at a cost. To authenticate once, the authentication information must be static, so that it can be reused without any user interaction. However, static authentication information, such as browser cookies or URL parameters, are easily accessed, extracted, and shared. Sharing and re-using static authentication credentials breaks the link between a user and their digital identity.

These authentication problems make ZT impossible. We must assume that attackers inside our network can steal static authentication tokens, so we cannot trust these tokens. Authentication must be dynamic and per-connection for ZT to work. Hence, SSO, as currently practiced, is not compatible with a ZT approach.

Can Segmentation Be Used in a ZT Architecture?

Segmentation involves breaking a network into segments and isolating the segments from each other. Taken to its logical extreme, often called *micro-segmentation*, this approach mimics ZT by protecting each individual resource within its own private segment. However, in practice, segmentation creates groups of resources within the overall collection. This is better than a single segment, but worse than ZT.

The problem with segmentation is that the segments must be isolated from each other for security, but they also must interact for functionality. This is accomplished by setting up boundary protections. These protections break and scan encrypted traffic at the boundary, which prevents end-to-end security and makes ZT impossible. Without end-to-end security, we must trust one or more boundary scanners, which is antithetical to the ZT philosophy. Thus, segmentation, except in the extreme case of defining a new segment for each individual resource, is an incremental improvement in the status quo but is fundamentally incompatible with a true ZT implementation.

Can Federation Be Implemented in a ZT Architecture?

An enterprise often wishes to collaborate with another enterprise. For a ZT enterprise, two types of collaboration are interesting: a full ZT collaboration, and a collaboration where the partner only has authentication available. With a full ZT collaboration, each enterprise uses ZT principles for policy-based access to resources. All that is needed is a service that can translate credentials from one enterprise to the other. Identities and access claims are mapped from external to internal representations, and credentials are re-issued in the local format to provide access to external entities.

For identity-based federation, the ZT enterprise has two choices. One option is to create identity mappings that also include access rights and privileges. This is not scalable due to the need to create these tailored mappings for each individual in the federation agreement. A second option is to delegate access from local entities to external entities. In this case, each external individual still needs to be granted access, but such decisions are distributed among the individuals who are sharing the resources. The delegation approach requires additional local services and data stores to function, so it extends the notion of ZT beyond its core implementation, but it provides a way to integrate federation partners cleanly within the local enterprise and preserves ZT ideas.

How Can Security Scanners Operate in a ZT Architecture?

Current security best practice includes the use of security scanners. These look for patterns in data, behavior, or other aspects of the network or its traffic in order to automatically identify, document, and stop potentially malicious activity. The most capable scanners operate at the application layer and understand the protocols and data formats in use. This requires access to encrypted content, which requires breaking end-to-end secure connections.

In order to move security scanning to a ZT architecture (ZTA), the scanning must be moved to the endpoints. This allows the preservation of end-to-end security, which is a core feature of ZT. Scanners rely on trusted hardware to validate and host them. Unlike centralized scanners, these endpoint-based scanners must be implemented in software. After this challenge is addressed, benefits can be realized with this new approach. Instead of a fixed scanner for the entire enterprise, scanners can be tailored to the particular endpoints. This means endpoints with lower scanning requirements can be scaled back to save resources, and endpoints with higher requirements can be scaled up to provide better security than a one-size-fits-all solution. However, the main benefit is the preservation of end-to-end security properties that form the foundation of the ZT approach.

Do I Need Infrastructure with Zero Trust?

Moving to ZT is not a simple configuration change. It requires some infrastructure to get started and to support the necessary security functions. A starting point for understanding the infrastructure requirements is a set of five security principles: 1) know the players, 2) maintain confidentiality, 3) separate access and privilege from identity, 4) maintain integrity, and 5) require explicit accountability. The ZT infrastructure is designed to address these.

The key elements of infrastructure that can address the security principles are: 1) an attribute store that contains authoritative information used for access decisions, 2) a registration process for enterprise resources and their access policies, 3) a service to match attributes to access policies and respond to queries about whether an entity has a particular level of access to a particular resource, and 4) a set of user convenience services that provide visibility and maintainability of the rest of the infrastructure components. With these key elements of infrastructure, the process of building or migrating to ZT can begin.

Why Zero Trust, Why Now?

William R. Simpson and Kevin E. Foltz

Synopsis

Zero Trust (ZT) is a new way to structure security defenses to better defend our digital resources against attackers. It is not a product or a security tool, but a way to organize the resources and the tools we use to protect them. Instead of a network-based defense, which places protections at the network boundary, ZT is a resource-based defense that places protections at each valuable resource. This provides a better match to current threats by directly protecting what is being attacked, and it provides a more resilient defense against lateral movement within an organization. For the Department of Defense (DoD) at this time, the current defense builds upon a clear concept of the fortress approach. Many of the requirements are based on inspection and reporting prior to delivery of the communication to the intended target. The inspection and reporting requires a number of software tools to preclude malicious entities from conducting activities such as exfiltration of data, theft of credentials, blocking of services, and other nefarious activities. These inspections require decryption of packets, which implies that the defensive suite either impersonates the requestor or has access to the private cryptographic keys of the servers that are the target of communication. This approach has been repeatedly bypassed and defeated by advanced persistent threats. The network-based approach has been repeatedly broken, which shows that it has not been working for some time now. ZT offers a new approach to defend our networks and digital resources.

The Current Approach

The current approach to security creates clusters of resources within network boundaries. All resources within a network segment receive protection from a set of security tools located at the boundary (or front door) of that network segment. Computer network defense is defined as "Actions taken through the use of computer networks to protect, monitor, analyze, detect and respond to unauthorized activity within the enterprise information systems and computer networks" [1]. The current defense package assumes that the threat can be stopped at the front door, as shown in Figure 1. All traffic in the enterprise, both coming and going, is routed through this front door. The front door is often onerous enough that administrator back doors are made available [2] to bypass many of the security checks. These backdoors, in addition to credential theft and threat stack vulnerabilities, are often the target of exploits. One example is the recent SolarWinds attack [3].

The elements involved in implementing network and application defense are numerous and complicated. Functionality is provided by a wide range of appliances. This functionality may be for quality of service to the user or quality of protection to network resources and servers. These appliances are often placed in-line, and some require access to content to provide their service. The literature is confusing because offerings include multiple services under various titles such as multi-function firewalls or advanced defense systems. The fortress defense has spectacularly failed with breaches occurring daily. The appliances in the package do stop the current threats for a short period, but new threats materialize very shortly and once again defeat the fortress approach. Even with detection and mitigation, we have continued threat presence over long periods. The advanced approaches described here assume that the threat is present and in the enterprise at all times. Although this may not be true at any given time, it is certainly true at various times during operations.

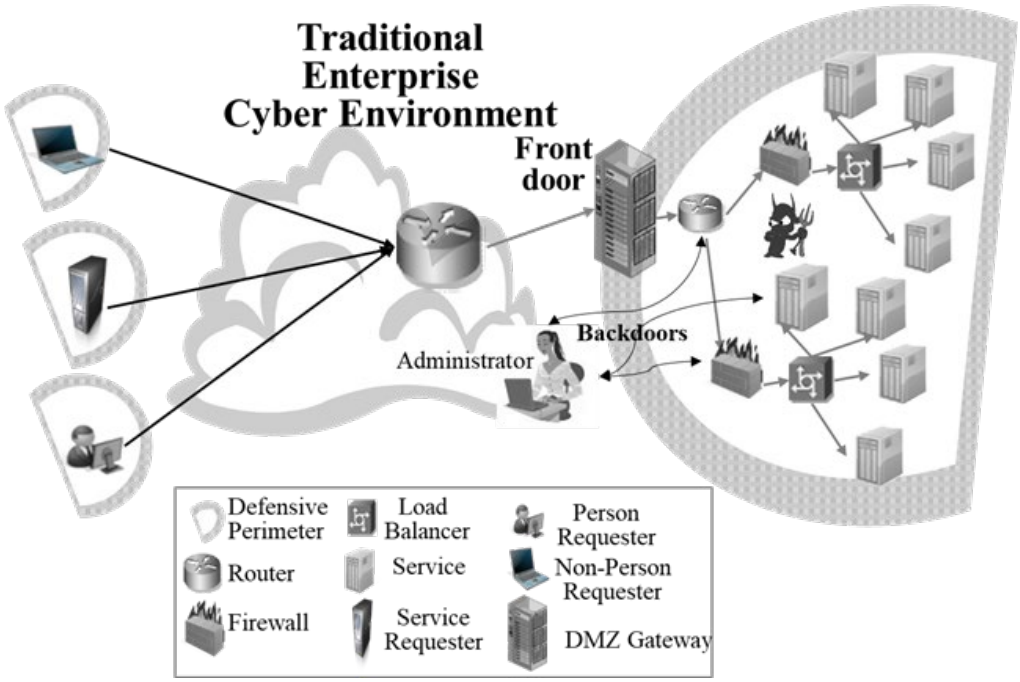

Figure 1. Fortress-Protected Enterprise

ZT

To fix the problems associated with network defense at the border, a new approach is needed. ZT is better suited to combating the current attack methods while preserving existing end-to-end security measures. ZT changes the one-size-fits-all security approach of a boundary defense to a custom-tailored approach for each resource within that boundary. The defenses are implemented at the resource, so there is no gap between the security and the resource it protects. ZT is an endpoint-based solution. It does not break the end-to-end secure communication channel between requester and resource. It scans at the endpoints and reports findings to a central monitoring facility. This allows requester and provider to authenticate each other directly and perform encryption and integrity from end to end. By focusing on the endpoints, ZT eliminates the man-in-the-middle (MITM) that boundary security introduces.

Many of the new security techniques have moved to a distributed security approach. The ZT framework is a distributed security system that eliminates or mitigates many of the primary vulnerability points inherent with the fortress system, as shown in Figure 2. Each entity needs assurance that the entity and device they are engaged with are known entities and, specifically, the ones to whom the communication should be allowed. However, it is this distributed approach and the requirement for content inspection and reporting that causes the conflict between this approach and the traditional fortress representation. All active entities and devices in ZT systems have public key infrastructure (PKI) certificates. Identity may be bolstered by using multi-factor techniques, and temporary credentials may be issued when necessary. Communication between active entities requires bilateral, PKI, end-to-end authentication of both the participants and their hardware.

ZT represents a change from current security practice. Instead of protecting resources by blocking outsiders, the protections are placed at the resources themselves. This approach is a better match to the current threats, which are consistently breaking through firewalls and other boundary protections. ZT provides defense against outsiders and malicious insiders, and it blocks attacker lateral movement within an enterprise.

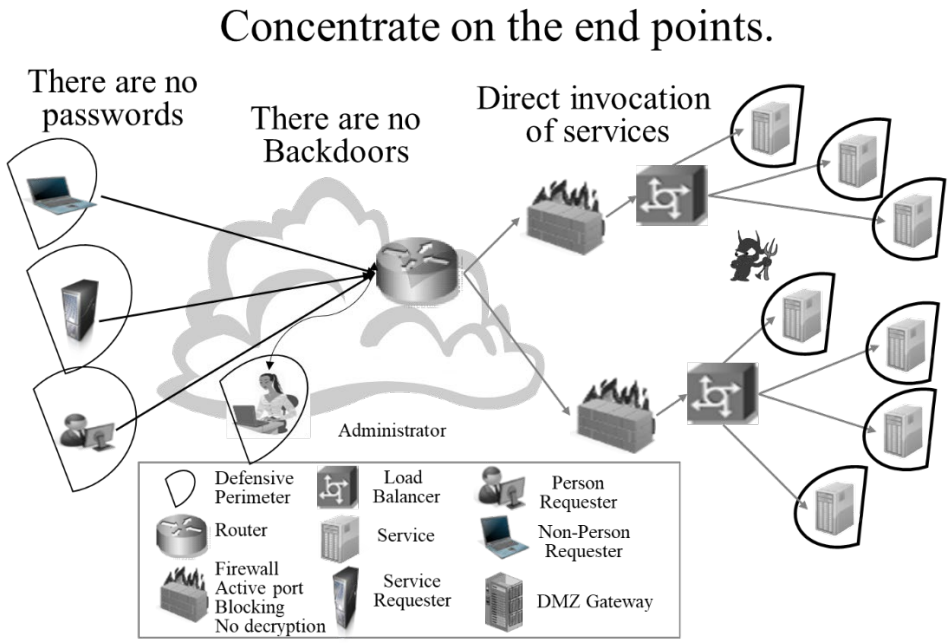

Figure 2. ZT Enterprise

To achieve this vision, we provide five foundational concepts for a ZT approach:

1. Two-way authenticated communication
2. Endpoint device management
3. End-to-end encryption and integrity
4. Policy-based authorization
5. Accountability for actions

In the DoD, these techniques have been fully developed, tested, and verified on the National Cyber Range and are described in the Air Force Consolidated Enterprise IT Baseline [4-6].

References

[1] Andress, Jason, and Steve Winterfeld. "Computer Network Defense." In *Cyber Warfare*, pp. 179–191. Rockland, MA: Syngress, 2011.

[2] TechTarget.com. "Backdoor (Computing)." https://searchsecurity.techtarget.com/definition/back-door, last accessed November 22, 2019.

[3] Datta, Pratim. "Hannibal at the Gates: Cyberwarfare and the Solarwinds Sunburst Hack." *Journal of Information Technology Teaching Cases*, March 12, 2021. https://doi.org/10.1177/2043886921993126.

[4] *Technical Profiles for the Consolidated Enterprise IT Baseline*, release 6.0. https://intelshare.intelink.gov/sites/afceit/ (CAC required).

[5] Simpson, William R., *Enterprise Level Security – Securing Information Systems in an Uncertain World*. Boca Raton, FL: Auerbach Publications, 2016.

[6] Simpson, William R., and Kevin E. Foltz. *Enterprise Level Security 2: Advanced Techniques for Information Technology in an Uncertain World* Abindgon, United Kingdom: Taylor & Francis Group, 2020.

Can SSO and ZTA Work Together?

William R. Simpson and Kevin E. Foltz

Synopsis

Single sign-on (SSO) is a convenience for users to avoid multiple authentication instances in computer-based sessions. It is a way to centralize authentication for a collection of related resources. It simplifies the process of authentication by providing users a single place to establish their identity, and a single method for resources to authenticate requesters. Zero Trust (ZT) Architecture (ZTA) is a security approach that moves protections away from network borders and to the resources themselves. It removes the ability and need to trust networks and requires each requester to prove access based on their credentials at the time of a request. The question is whether these two can work together. The short answer is "No," but the full answer is more nuanced because the term SSO is used somewhat loosely. We look at the concepts of SSO and ZTA and show how the most common use of SSO does not work with ZTA.

The SSO Approach

SSO transfers authentication information between endpoints. The SSO server creates an SSO token after a requester authenticates to the SSO server [1]. This authentication may be tailored to the resource the user is requesting, with multi-factor or other methods to provide different strengths of authentication. In addition, the SSO server may provide many different options to accommodate users with different credentials, locations, and devices. The primary motivation to adopt SSO is often ease of use. This applies to both the users and the enterprise. The users have a single portal for authentication that accommodates all users, and the enterprise implements one authentication server and simply implements token processors at the resources. It is centralized, efficient, and easy to use.

However, SSO is typically not secure. Any authentication token that can be reused or transferred between users allows impersonation, a fundamental violation of basic security. SSO tokens are often implemented as "bearer tokens," meaning that the bearer (whether a proper user or attacker) can use the token to authenticate as the associated requester. SSO tokens are protected by Hypertext Transfer Protocol Secure (HTTPS) from SSO server to requester and again from requester to resource, but this piecemeal security leaves a gaping hole at the requester. Tokens that are implemented as a URL parameter or a cookie in the HTTP header can be easily copied and shared among users. The SSO approach is better than no security, but it falls short of the Department of Defense's (DoD) needs, and the complexity of proper implementation means a one-size-fits-all approach will cater to the lowest security level of the systems it supports.

The ZTA Approach

ZTA, based on NIST 800-207 [2], is built on a set of tenets that include the following:

- Access to individual enterprise resources is granted on a per-connection basis
- Entity authentication is dynamic and strictly enforced before access is granted

The first tenet implies that each connection must be authenticated. For a requester to connect to a resource, the requester must authenticate to that resource as part of the initiation of the connection. For web requests, this is typically accomplished through HTTPS. When users have a common access card (CAC), the authentication can be done natively within the transport layer security (TLS) protocol that provides security for HTTPS traffic. In other cases, the server credential is used to establish an encrypted connection with integrity protection, and the user authenticates within this connection prior to requesting and receiving access to the server resources. The first method is preferred because it ensures that only two-way

authenticated communications take place, but the second method, if strictly enforced, can offer a comparable substitute.

The second tenet mentions dynamic authentication, which implies that the user must do something to prove their identity in an active way. Passive authentication would rely on the use of old credentials or session information, or it may reference a prior authentication rather than performing the full authentication again. For ZTA, such passive measures are not sufficient. If a user requests a resource, it must authenticate at the time of each request and in association with each connection. Relying on prior authentications by reference allows an attacker to do the same, which bypasses actual authentication in favor of simplicity or performance.

A Fundamental Incompatibility

SSO authenticates on one connection and provides resources on another connection. This violates the first ZTA tenet. Although SSO authentication to the SSO server is dynamic and may be strictly enforced, the access is being granted at the resource, and the resource only receives a static SSO token, not a dynamic, interactive authentication. This violates the second ZTA tenet.

The problem is that the SSO token provides no guarantee that the holder of the token is the entity named in the token. It is a bearer token. Thus, security relies on externally trusted entities, policies and practices. This is not the ZT approach.

Some Nuances

SSO is a broad term that can mean many things, and some implementations are better than others. However, the key problem for ZTA is the reliance on trust of external elements. One is the user. A user can easily extract, copy, and share the SSO token received from the SSO server. If a user can do it, an attacker can do it too. Often, the attackers are better at this than most users, and stopping these attacks can be difficult due to the contrasting requirements for security and maximum functionality in browsers and web protocols.

It is possible to argue that all authentication is essentially SSO. For example, even with a CAC, the true authentication occurs when the user presents their paperwork and identification at a CAC office. The CAC itself is simply a glorified SSO token. There is some truth to this. However, the difference is that sharing a CAC generally means that the original user is giving up their CAC, and such sharing would be easier to detect if an attacker stole a CAC due to its uniqueness in hardware. Software-based SSO tokens issued by a server can quickly and easily be duplicated and widely shared without a user's knowledge. Thus, the CAC can conceptually be thought of as SSO, but it provides much tighter security than the standard SSO solutions.

Another middle ground is multi-factor authentication (MFA). With MFA, a user may combine multiple different types of authentication to receive access. If done directly with the resource provider, this is a strong security approach. An alternative proposal is to issue a public key infrastructure (PKI) credential after MFA that can be used much like a software-based CAC. This is less secure due to the software nature of the credential, but it is issued with a short expiration, such as 90 minutes, instead of a year or more. The key difference here is that the user does not share the full credential with the resource. It protects and uses the private key to provide a signature that validates ownership of the associated public key. The connection between the credential issuer and requester must be secured, but there is less risk of replaying authentication information between the requester and resource when compared with standard SSO solutions.

In reality, SSO is a spectrum. On the strong end is the CAC, which can be thought of as a hardware-based non-shared SSO token, followed by a temporary PKI credential, which is a software-based, short-term, non-shared SSO token. On the weaker end are various approaches that produce a shared token transmitted by

the browser. The token is what allows authentication, and the token is sent in its entirety to the resource, which opens up a large attack surface. Short-lived SSOs reduce the attack surface time. There are additional considerations within the weaker end of SSO, such as whether the token is encrypted or signed. Encryption prevents a third-party observer from knowing the contents of the token, which might prevent them from knowing how to use (or abuse) it. A signature on a token can be used to maintain its integrity. Tokens that can be parsed without a signature allow an attacker to capture a token and modify it. The receiver cannot distinguish such a token from a valid token, so a token without a signature enables escalation of privilege for valid and invalid users. However, regardless of whether a token is signed or encrypted, the ability to share and replay a token for access is the main security vulnerability associated with normal SSO.

Summary

SSO offers a spectrum of choices where an initial strong authentication results in a weaker temporary form of authentication that can be used at resources. CAC or PKI credential issuance are stronger forms of this approach that allow the requester to retain secret keys. The standard SSO practice of issuing software tokens for access is weaker because it may allow replay, sharing, or modification. The commonly used SSO approaches that use one server to issue tokens and another server (or module) at the resource to parse the token is weak SSO and is not suitable for a ZT environment where authentication is dynamic and strictly enforced for each connection. ZT is about reducing trusted elements in the network. Requiring trust of the token issuer, the user, and the user's hardware and software goes against the simple notions inherent in ZT that call for dynamic and strictly enforced authentication on each connection. In short, SSO and ZTA cannot work together.

In the DoD, these techniques have been fully developed, tested, and verified on the National Cyber Range and are described in the Air Force Consolidated Enterprise IT Baseline [3–5].

References

[1] Teravainen, Taina. "Single Sign-on (SSO)." https://searchsecurity.techtarget.com/definition/single-sign-on, last accessed April 26, 2021.

[2] Rose, Scott, Oliver Borchert, Stu Mitchell, and Sean Connelly, *National Institute of Standards and Technology (NIST) SP 800-207, Zero Trust Architecture*. Washington, DC: National Institute of Standards and Technology, August 2020.

[3] *Technical Profiles for the Consolidated Enterprise IT Baseline*, release 6.0. https://intelshare.intelink.gov/sites/afceit/ (CAC required).

[4] Simpson, William R., *Enterprise Level Security – Securing Information Systems in an Uncertain World*. Boca Raton, FL: Auerbach Publications, 2016.

[5] Simpson, William R., and Kevin E. Foltz. *Enterprise Level Security 2: Advanced Techniques for Information Technology in an Uncertain World* Abindgon, United Kingdom: Taylor & Francis Group, 2020.

Can Segmentation Be Used in Zero Trust?

William R. Simpson and Kevin E. Foltz

Synopsis

Within the Department of Defense (DoD), network defense utilizes a comprehensive set of hardware and software tools to preclude malicious entities from conducting nefarious activities. Most current enterprises build their defenses upon a fortress approach. Network defense tools defend this fortress, which defines a clear boundary between the untrusted outside and the trusted inside. Network segmentation expands on the fortress idea to create a layered fortress model, where a larger fortress consists of smaller fortresses with their own boundaries and protections. This provides more layers of defense, which limits threat mobility and helps to contain damage during exploits and intrusions. Zero Trust (ZT) starts with a different model, where the individual resources are protected, and there is no reliance on the network for protection. This has the same goals of limiting threat mobility and containing damage. Although network segmentation shares similar goals with Zero Trust Architecture (ZTA), it has fundamental incompatibilities that prevent it from being a useful security enhancement within a ZTA.

Network Segmentation

Network segmentation is a term for dividing a network into multiple subnetworks, or segments, and managing access to these segments. Typically, it involves segregating traffic between the network segments and enforcing segment policies with firewalls or other security appliances. A typical segmentation is shown in Figure 1. Segmentation may involve the use of physical sub-networks or Virtual Local Area Networks (VLANs).

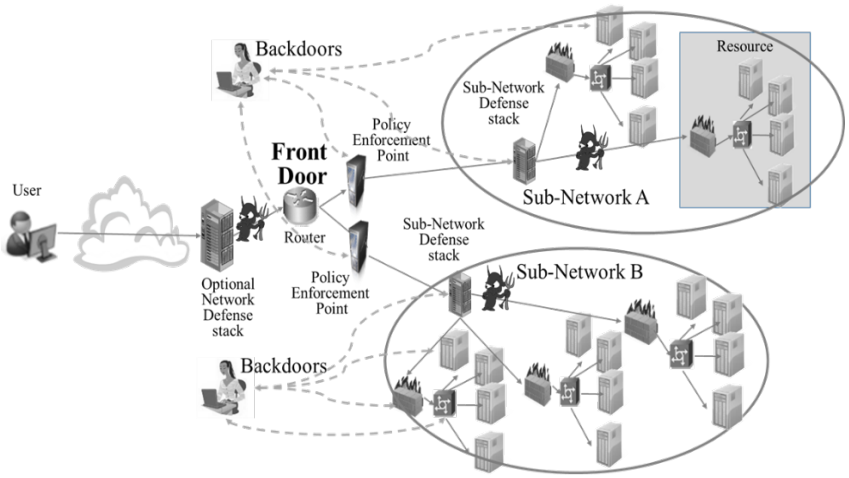

Figure 1. Segmented Network

The degree of network segmentation is determined by two things: the separation of resources into different segments, and the grouping together of resources within the same segment. The terms *macro segmentation* and *micro-segmentation* qualitatively describe different ends of this spectrum. With extreme macro-segmentation, we arrive at the fortress approach for the entire enterprise. With extreme micro-segmentation, we arrive at endpoint defense, where each endpoint is treated as its own fortress. Most real world implementations fall between these extremes and involve a number of segments that each contain a number of resources. Micro-segmentation reduces lateral movement of threats and provides more granular access by allowing different policy rules for each of the segments. Resources may be protected

by appliances for both the segment and/or the whole network. With complicated or diverse resource access policies, it may be very difficult to implement a meaningful segmentation security policy. A fundamental problem for segmentation is based on its reliance on the fortress approach for security. It is still based on the flawed assumption that a robust front door can prevent attacks from outside. Attackers find ways through these protections, and they will move laterally within segments, just as they did for the fortress. Segmentation increases the complexity and must be carefully configured. Any misconfiguration is a new vulnerability.

ZTA

The ZTA paradigm addresses lateral threat movement within the network and moves defenses from network-based perimeters to focus on users, assets, and resources [1]. Each entity in a communication must have assurance that the party they are engaged with is a known entity and, specifically, the one to whom the communication is intended. Access and privilege should only be granted to an authenticated identity if credentials for access and privilege are presented, verified, and validated. Finally, all communications should be encrypted and provided with integrity protections that allow the recipient of communications to verify that what was received was actually sent. [2] and [3] provide extensive descriptions of these processes. Moving to ZTA requires an assessment of the benefits versus the risks. Moving from a single boundary defense to multiple resource defenses allows increased flexibility. Each resource can tailor its defenses to its own needs. However, this increased complexity, if not properly managed, can introduce its own vulnerabilities.

Combining Segmentation and ZTA

We first consider a full security implementation of both approaches. We then examine a hybrid solution that mixes parts of each. Finally, we consider non-security benefits.

A. Full Security Combination

First, we consider implementing segmentation on an existing ZTA. ZTA requires seamless end-to-end encrypted communication for active entities. Segmentation adds boundary security components that must break end-to-end security to view network traffic. These boundary components are passive entities in ZTA. As passive entities, they do not have the ability to decrypt network traffic, and they cannot perform their functions. Thus, segmentation cannot help an existing ZTA without breaking ZTA security. Combining segmentation and ZTA results in problems from a security perspective. The key issue is determining how to handle secure communication at segment boundaries: Segmentation requires breaking it and ZTA requires preserving it. Because of this fundamental difference, it is not possible to fully implement both approaches in the same enterprise.

B. Hybrid Approach

A full implementation of both approaches does not work, but when segmentation is finely applied such that each segment is a micro-segmentation, conditions essentially match a full implementation of ZTA. Micro-segmentation of the individual resource together with an embedded network defense stack preserves the end-to-end communication path. This association of micro-segmentation and ZTA provides the basis for a hybrid solution.

Areas of micro-segmentation within an overall segmentation that includes both micro- and macro-segmentation can be converted to a local ZTA solution. This conversion of a single segment to ZTA can be applied to all regions of micro-segmentation. Figure 2 illustrates a hybrid enterprise segmentation using macro- and micro-segmentation. Using ZTA on the micro-segmentation paths and normal defense in depth on the macro-segmentation provides the overall hybrid solution. Note that although the backdoors persist in the normal segmentations, they are eliminated in the ZTA architecture, and

administrators and other previous exceptions must go through the front door for connection. This is less onerous for administrators as they have an unbroken and direct encrypted connection to the endpoint they seek. Converting additional parts of the macro segmentation into micro-segmentation results in a migration path from fortress to ZTA using segmentation.

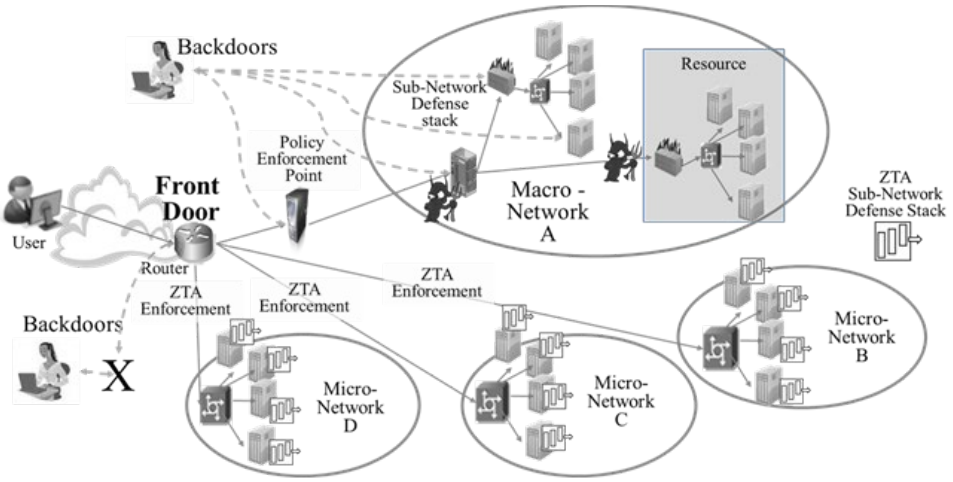

Figure 2. Combine Micro/Macro Segmentation for ZTA Transition

C. Other Considerations

Although segmentation and ZTA cannot be fully combined for security, dividing network traffic between different segments may reduce the aggregate network traffic on each segment, which improves performance. Use of virtual local area networks (VLANs) instead of hardware can offer cost savings and improved configurability. Software-defined networks can improve network traffic performance. These segmentation benefits do not require breaking encryption at boundaries and show that although segmentation does not meet ZTA security requirements, it can provide other benefits.

In DoD, zero tolerance techniques have been fully developed, tested, and verified on the National Cyber Range and are described in the Air Force Consolidated Enterprise IT Baseline [4-6].

References

[1] Rose, Scott, Oliver Borchert, Stu Mitchell, and Sean Connelly, *National Institute of Standards and Technology (NIST) SP 800-207, Zero Trust Architecture*. Washington, DC: National Institute of Standards and Technology, August 2020.

[2] RSA Laboratories. "PKI Standards: PKCS#12 format PKCS #12 v1.0: Personal Information Exchange Syntax Standard." PKCS 12 Technical Corrigendum 1, RSA Laboratories, February 2000. http://www.rsa.com/rsalabs/node.asp?id=2138.

[3] Cantor, Scott, John Kemp, Rob Philpott, and Eve Maler (Eds.). "Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0." OASIS Standard, March 2005

[4] *Technical Profiles for the Consolidated Enterprise IT Baseline*, release 6.0. https://intelshare.intelink.gov/sites/afceit/ (CAC required).

[5] Simpson, William R., *Enterprise Level Security – Securing Information Systems in an Uncertain World*. Boca Raton, FL: Auerbach Publications, 2016.

[6] Simpson, William R., and Kevin E. Foltz. *Enterprise Level Security 2: Advanced Techniques for Information Technology in an Uncertain World* Abindgon, United Kingdom: Taylor & Francis Group, 2020.

How Does Federation Impact Zero Trust?

William R. Simpson and Kevin E. Foltz

Synopsis

Federation involves information sharing among services and with working partners, coalition partners, first responders, and other organizations. Federation may be unilateral or bilateral with similar or dissimilar information-sharing goals. Federation partners may support the same security policies with compatible standards and services, a similar but incompatible security framework, a subset of required security services, or no security services. Federation partners can only support a full Zero Trust (ZT) implementation in two ways: ensuring access by identity or by using the same security policies with compatible standards and services. All implementations compromise security and increase vulnerability and risk.

ZT Federation Introduction

ZT and federation might seem contradictory at first. ZT implies that you do not trust even your own organization, and federation implies trusting another organization. However, it often arises that one organization, or part of it, desires to share resources with entities in another organization. In some cases, the trust is informal and based on personal relationships between members. In other cases, the two organizations are closely related but logically separated by the use of different security architectures.

The two practical challenges for federation are breaking down the existing walls and integrating the different security architectures of the two organizations. The first challenge may require bypassing network scanners and adding endpoint security. This would be a solution for a partner that is not fully ZT. Selective bypassing combined with endpoint-based security scanners would allow ZT to work for the federation partner while other connections would continue to be scanned at the border. The second challenge is to translate between security implementations within the two federation partners. If the security is close enough, rules can be defined to perform this translation automatically. As security implementations diverge, more up-front and ongoing effort is required to bridge the difference.

Federation Technical Consideration

Some of the technologies and principles of federation are listed below. This section discusses different ways to extend the principles of ZT to a federation partner based on the partner's technical capabilities. Options include:

ZT Compatible:

- ZT federation
- Identity-based federation

Non-ZT:

- ZT-like federation
- Weak identity federation
- Ad hoc federation
- Person-to-person federation

The federation options are listed from most to least technologically compatible. This paper discusses only the ZT-compatible solutions. All federation types are discussed in [1].

ZT Federation

ZT federation is an agreement to accept identity and access claims from another enterprise. Federation is a long-term substantial agreement that is made at the enterprise level. To resolve federation issues, the federation service relies on the following information:

- Certificates of federated identity claims for validating authentication public keys and chaining to a trusted root certification authority (CA).
- Certificates of federated access claims for validating signatures and chaining to a trusted root CA.
- A set of identity-mapping pairs with the form (Identity1, Identity2), in which Identity1 is a token issued by the federated access claims service and is to be mapped to Identity2 in the local enterprise.
- A set of mapping pairs of the form (Claim A, Claim B), in which Claim A is issued by the federated access claims service and is to be mapped to Claim B in the local enterprise.
- Additional attribute mappings associated with claim mappings.

The federation service is a translator for access claims. It receives a valid access claim from a federation partner and creates a local access claim based on the federation information provided above. Local applications and services provide access based on the translated token. This preserves the enterprise-wide nature of federation and does not require local changes to support federation agreements.

An example of data captured in federation agreements is shown in Table 1. This shows the data for two separate federation agreements. Each web service in the enterprise has a limited number of trusted root CAs for authentication credentials and access claims signatures stored in its trust store. With ZT federation, a list of trusted partner CAs and federation services is established. Trusted partner CAs for identity credentials are distributed to applications and services. Trusted federated access claims credentials are distributed to federation services within the enterprise. The federation service is called by a service provider when an unknown access credential is encountered; the federation server checks against known federation partners to validate the credential, creates a new authorization token with its own signature, and returns this to the application or service for processing.

For identity and access claim mappings, the special cases of “null” and “no change” are acceptable in addition to explicit values. “Null” removes the claim or identity, whereas “no change” leaves the original claim or identity. Access claims in the federation partner token must match the federation agreement exactly. Access claims in the re-issued token must match access claims for the target application or service. Identity and access claim mappings are added to the federation store after an amendment to the federation agreement. Revocation of a federation agreement is accomplished by removing the federation partner from the trusted federation data store. When a federation service recognizes and validates a partner authorization credential, it uses its mapping list to map the received credential into a new credential with possibly different identity, access claims, and attributes. This new credential is signed by the federation service and returned to the requesting application or service. The application or service then processes this new token as though it had received it from a valid requester within the enterprise. Failure to validate an incoming token by the federation service results in an error message response to the application or service, which leads to an authorization failure at the application or service.

Table 1. Federation Data Requirements

Federation Partner 1 Information		
Certificate	**Federation Partner 1 certificate and chain to root CA**	
Identity Mappings	**Identity 1**	**Identity 2**
	Identity A	**Identity B**
	Identity r	**<no change>**
	…	**…**
Claim and Attribute Mappings	**Claim A**	**<null>**
	Claim n	**Claim z**
	Claim y and Attribute q	**Claim y and Attribute r**
	…	**…**
Federation Partner 2 Information		
Certificate	**Federation Partner 2 certificate and chain to root CA**	
Identity Mappings	**Identity x**	**Identity y**
	Identity Q	**Identity R**
	…	**…**
Claim and Attribute Mappings	**Claim n**	**<no change>**
	Claim p with Attributes x, y, z	**Claim p with Attribute k**
	Claim A	**<no change>**
	…	**…**

Identity Credential Federation

A federation partner may provide identity credentials but not access credentials, such as an account-based system in which the ID is used to log in. In this situation, a basic function of ZT is missing, so ZT federation is not possible, but the identity credentials can still be used as a starting point for federation. One solution is a mapping from partner identity to local identity that also includes associated access claims. This would be performed at the federation service as part of the federation mappings. This is not the typical use for these mappings, and it requires modifications to Table 1 to combine the identity, claim, and attribute mappings. This method would work if the identity credential is passed for authentication and if the authorization information is exchanged between enterprises and incorporated into the federation agreement. This approach does not scale well, and it should only be used for small sets of requesters. Federation mappings are intended to be for claim and attribute equivalences and generic identity transform rules, not explicit per-entity claim and attribute information.

For larger-scale federation, the data owner or some other entity with access to the data can delegate the appropriate access claims to the appropriate individuals in the partner enterprise. Delegation is not normally part of ZT, but is described in [2] and has been implemented in the consolidated Enterprise IT Baseline [3]. The delegation framework within ZT is designed to allow such short-term access to specific individuals. In this case, the delegation service, rather than the federation service, maintains the mappings. This is a more appropriate place to store this information and is scalable due to the distributed nature of the different individuals managing different delegation rules. The basic structure for federated delegation is shown in Figure 1. The dashed lines represent the flow for setting up the delegation. The local delegator uses the special delegation service to assign access claims to the federated partner identity as stored in the authoritative content store for federation partners. The special delegation service is separate from the standard in-enterprise delegation service, and the content store for federation partners is separate from the normal store for in-enterprise entities. This special delegation service uses the access claims exposure and editor service to store access claims for the federated identity in the access claims

repository. The solid lines represent the flow for a federated requester to retrieve a token with delegated access claims. The partner interfaces with the federated access claims service, which issues access claims according to the delegation rules associated with the partner's identity.

Unlike a local entity, federation partner identities are stored in a content store for federation partners. However, all access claims, for both local and partner entities, are stored in the claims repository. This provides a seamless method to retrieve access claims while maintaining separation of identities.

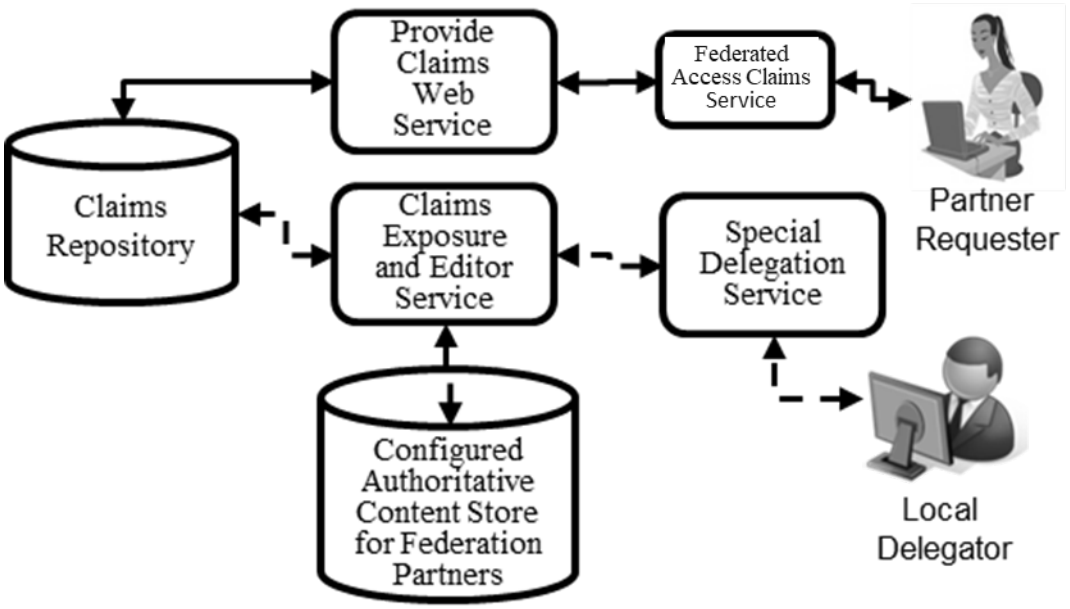

Figure 1. Identity Credential Federation using Delegation

Summary

Federation of a ZT enterprise with another enterprise will affect the security of the original ZT enterprise. Depending on how closely matched the two are, changes may be needed to enable federation while preserving ZT properties. Two variants in particular can be made to work. The first is where ZT concepts are present within the federated partners. In this case, the basic security structures remain in place, and mappings are made between them. The second is through identity credentials using a method called delegation, which is not a normal part of ZT. Upon the addition of delegation, the normal ZT security process remains largely unchanged under federation.

In DoD, ZT techniques have been fully developed, tested, and verified on the National Cyber Range and are described in the Air Force Consolidated Enterprise IT Baseline [3-5].

References

[1] Simpson, William R., and Kevin E. Foltz. "Maintaining Zero Trust with Federation," *International Journal of Emerging Technology and Advanced Engineering* 11, no. 3 (March 2021): in Press.

[2] Foltz, Kevin E. and William R. Simpson. "Delegation of Digital Access and Privilege in a Secure Enterprise" In *Proceedings of The 22th World Multi-Conference on Systemics, Cybernetics and Informatics: WMSCI* 1: 125–132. Orlando, FL: July 8–11, 2018.

[3] *Technical Profiles for the Consolidated Enterprise IT Baseline*, release 6.0. https://intelshare.intelink.gov/sites/afceit/ (CAC required).

[4] Simpson, William R., *Enterprise Level Security – Securing Information Systems in an Uncertain World*. Boca Raton, FL: Auerbach Publications, 2016.

[5] Simpson, William R., and Kevin E. Foltz. *Enterprise Level Security 2: Advanced Techniques for Information Technology in an Uncertain World* Abindgon, United Kingdom: Taylor & Francis Group, 2020.

How Can Security Scanners Operate in a ZTA?

William R. Simpson and Kevin E. Foltz

Synopsis

Security scanners are a central feature in modern enterprise architectures. Email can contain malicious attachments or links to malicious content. Web pages may contain malicious content. Infected hosts may perform malicious activity within the enterprise. Security scanners look at the communications and their content to attempt to detect and stop malicious activity. They require some of the most trusted positions within the enterprise network to perform their duties. This contrasts the basic premise of Zero Trust (ZT), which asserts that the network is not to be trusted. This paper provides a way to resolve this apparent conflict by moving scanner capabilities from central locations and to the endpoints. This eliminates centralized scanners as attack targets and removes the need to trust these central entities in the network, making security scanning consistent with the ZT approach.

The Current Security Scanner Approach

Security scanners typically operate on network traffic. In some cases, the scanner resides where network traffic naturally converges, such as a firewall or gateway router. In other cases, traffic is explicitly routed to the scanner before it is sent onward to its destination. The first approach works well for traffic between enclaves, and the second works better for traffic within an enclave. The scanner breaks the end-to-end encryption of the communication, scans traffic that has already been decrypted, or simply scans unencrypted traffic as-is. A notional setup of traffic scanners is provided in Figure 1.

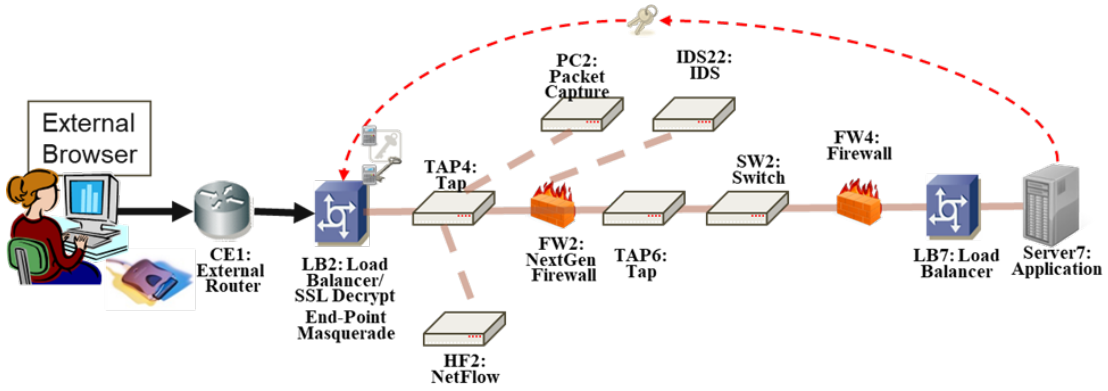

Figure 1. Notional Setup of a Scan Architecture

In a secure enterprise, traffic is encrypted, so scanning requires breaking this encryption. This can be accomplished by another entity, such as a load balancer (as shown in Figure 1) or a firewall, which integrates with the security scanner. In this case, the scanner is positioned within or near the other entity so that it can operate on the unencrypted data. Scanners can also sometimes break encryption themselves, which removes the requirement for any other entities and allows more flexibility of placement. In either case, as the traffic is decrypted between the two endpoints, the two endpoints must trust this man-in-the-middle (MITM).

The Zero Trust Architecture Approach

ZT Architecture (ZTA), based on NIST 800-207 [1], is built on a set of tenets, which include the following:

- All communication is secure regardless of network location.
- Entity authentication is dynamic and strictly enforced before access is granted.

The first tenet implies that all communication must be encrypted and protected for integrity. This includes both external and internal communication paths. Encryption ensures that intermediate nodes cannot view the contents being exchanged. Although metadata such as addresses, protocols, and protocol details may be visible, the application layer content is protected using encryption. The encryption methods can be negotiated with Transport Layer Security (TLS) as part of the HTTPS protocol. Integrity protections can also be negotiated through TLS and can either be incorporated within the encryption algorithm or implemented separately and added as a message authentication code (MAC) or through some other method. The secure enterprise approach with end-to-end encryption is shown in Figure 2. Passive elements are not allowed to manipulate contents.

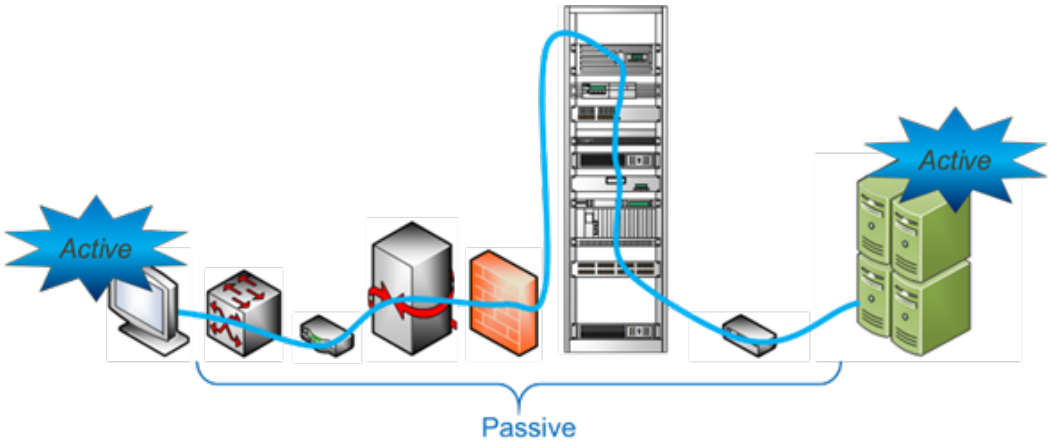

Figure 2. End-to-end Security Is Performed at the Endpoints Instead of Within the Network

The second tenet implies that each connection must be authenticated, and the user must do something to prove their identity in an active way. This does not permit third party authentication or MITM approaches. The two entities that are communicating must actively authenticate each other.

Resolving the Apparent Incompatibility

Security scanners, as currently implemented, violate both of the ZT tenets above. They prevent end-to-end secure communication by explicitly breaking these connections and scanning the contents. The security scanners act as the MITM, which is not permitted by ZT. The MITM scanners also prevent dynamic authentication. The endpoints can only authenticate to the MITM and can only authenticate the MITM. As a requester, there is no way to know whether content from the MITM accurately reflects the data that the actual source provided, or whether the MITM even retrieved the data from the intended source or just generated the data itself.

Although most scanners are benevolent, they are not attack-proof. Every piece of hardware and software has vulnerabilities, and security scanners are no exception. A compromised security scanner acting as the MITM is particularly dangerous because an attacker can potentially view or modify any traffic in the enterprise in arbitrary ways. This is why the ZT approach necessitates not trusting the network. The assumption is that attackers are in the network already, which may include the network security scanners.

A new approach to security scanning is required for ZT. This approach must be consistent with endpoint-based security, because ZT protects the resources, which are located at the endpoints. This new approach has three key features:

- Implement security scanners in software instead of hardware.
- Move scanners from the network to the endpoints.
- Tailor the scanners to the resources they protect.

The first feature enables the other two. Hardware boxes must be physically placed in a location on the network. They are expensive, high-performance, and difficult to duplicate or otherwise scale up or down. Hardware-based protection leads to a centralized approach due to the characteristics of the hardware itself, which is not consistent with ZT.

The second feature uses the software-based scanners to provide protection where it is needed: at the resource endpoints. These endpoints integrate the scanner software into their existing software at a point where the content to be scanned is available. If multiple layers are to be scanned, the scanner software can access multiple points of the processing pipeline, from raw network traffic to internal application data. No extra decryption or authentication is needed, because the scanner is now part of the endpoint itself instead of a separate entity. Although this does not eliminate the threat of compromise of this code, such a compromise now only affects one resource rather than the entire enclave.

The third feature uses the modular nature of individual software scanners to tailor the protections at any particular endpoint. Instead of implementing a stack of security scanners that is the same for all traffic, the scanners for a particular endpoint can be selectively implemented. An email resource uses email scanners, and a web server uses web traffic scanners. This reduces the performance requirements for the scanners because they are only scanning relevant traffic instead of all network traffic. Also, higher security resources could utilize a full security scanner stack, whereas lower security resources could selectively utilize a smaller set.

A notional server setup, providing end-point scanning is depicted in Figure 3. The migration path from the current approach to the ZT approach is fairly simple, but the benefits are only realized with a full transition. The initial transition can move scanners one-by-one from a central position to the endpoints. However, until all scanners are at the endpoints, the central MITM must remain, which negates ZT principles. Other benefits, such as performance, scalability, and tailoring of protections to resources can be achieved with a partial approach, but the core ZT ideas are only realized when all scanners are moved to the endpoints and the central MITM is removed.

Summary

Security scanners currently operate as the MITM at network choke points. They scan aggregated network traffic by breaking the secure connections and analyzing the unencrypted content being transmitted between endpoints. ZT does not allow such behavior, so a new approach based on endpoint scanners is recommended. Centralized hardware-based scanners are re-implemented as endpoint-based software scanners, which are then tailored to the endpoint protection needs. This preserves the scanning capabilities in a way that is consistent with ZT principles.

In the Department of Defense (DoD), these techniques have been fully developed, tested, and verified on the National Cyber Range and are described in the Air Force Consolidated Enterprise IT Baseline [2-4].

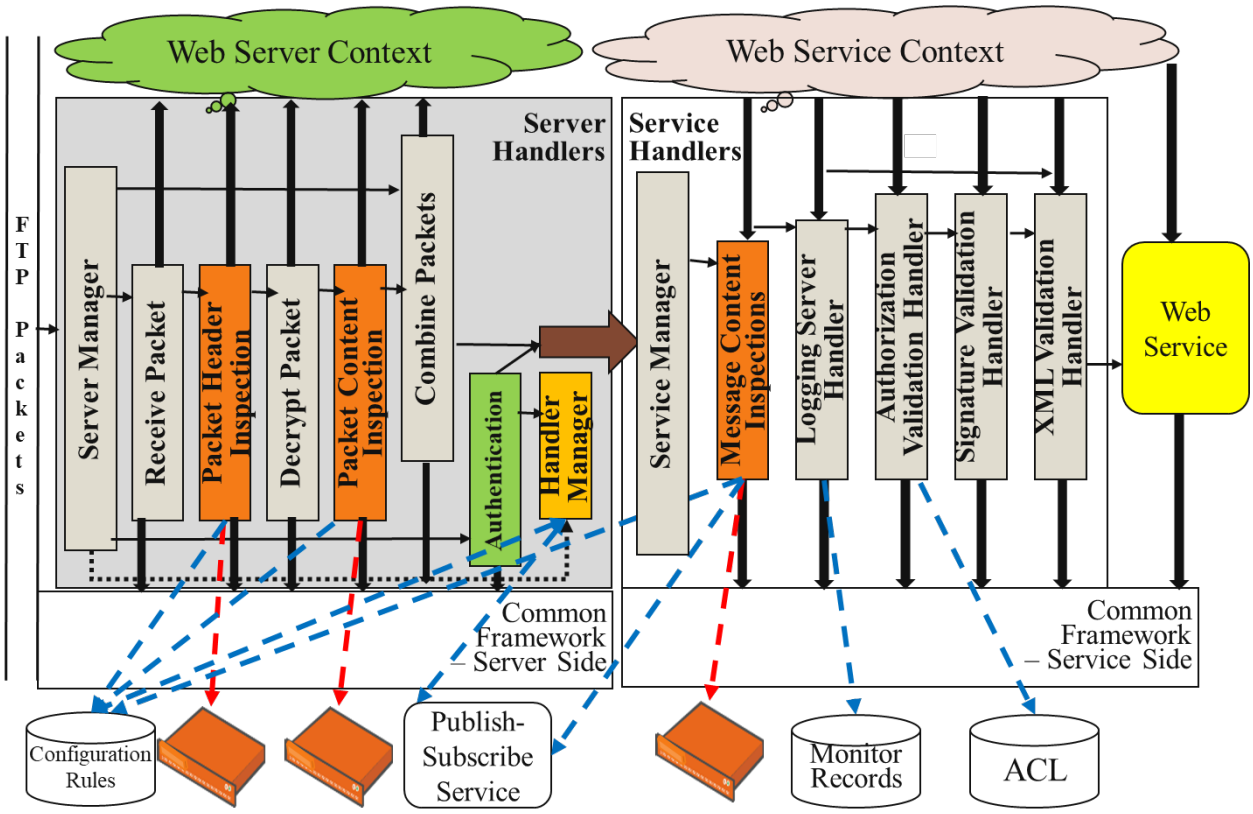

Figure 3. Notional Request Processing at the Server

References

[1] Rose, Scott, Oliver Borchert, Stu Mitchell, and Sean Connelly, *National Institute of Standards and Technology (NIST) SP 800-207, Zero Trust Architecture*, August 2020.

[2] *Technical Profiles for the Consolidated Enterprise IT Baseline*, release 6.0 available at https://intelshare.intelink.gov/sites/afceit/ (CAC required)

[3] Simpson, William R., *Enterprise Level Security – Securing Information Systems in an Uncertain World*, Auerbach Publications, May 2016, ISBN 9781498764452, .

[4] Simpson, William R. and Kevin E. Foltz, *Enterprise Level Security 2: Advanced Techniques for Information Technology in an Uncertain World*, Taylor & Francis Group, September 2020, ISBN 9781003080787.

Do I Need Infrastructure with Zero Trust?

William R. Simpson and Kevin E. Foltz

Synopsis

Zero Trust (ZT) is a way to structure security defenses to better defend our digital resources against attackers. It is not a product or a security tool, but a way to organize the resources and the tools we use to protect them. The main change is to put protections at the resources that need protection. The enterprise must enforce the use of identity credentials for authentication, in which the credential issuer is trusted, and credentials are verified and validated. The enterprise eliminates obsolete credentials, updates all entity attributes, maintains multi-factor data for users, and creates new credentials. The providers of the system's servers and services must be trusted entities within the enterprise. For large systems some infrastructure must be maintained for trusted individuals and their attributes. Most Department of Defense (DoD) organizations need an infrastructure to implement ZT. The "fortress approach" used by large enterprises (like DoD) has been broken for a long time and has been repeatedly bypassed and defeated by advanced persistent threats. ZT offers a new approach to fix these problems.

Zero Trust

To fix the problems associated with network defense at the border, a new approach is needed. ZT provides a more promising approach to combat the current attack methods while preserving existing end-to-end security measures. ZT changes the one-size-fits-all security approach of a boundary defense to a custom-tailored approach for each resource within that boundary. The defenses are implemented at the resource, so that there is no gap between the security and the resource it protects. ZT is an endpoint-based solution that does not break the end-to-end secure communication channel between requester and resource. It scans at the endpoints and reports findings to a central monitoring facility. This allows requester and provider to authenticate each other directly and perform encryption and integrity from end to end. By focusing on the endpoints, ZT eliminates the man-in-the-middle (MITM) that boundary security introduces.

Background

To achieve ZT, we provide five foundational concepts for a ZT approach. The minimal capability instantiation will provide bilateral mutual authentication and a claims-based system for access and privilege [1]. The ZT approach addresses five security principles that are derived from the basic tenets:

1. Know the Players: In ZT, this is done by enforcing bilateral end-to-end authentication using Public Key Infrastructure (PKI) certificates issued by an enterprise approved Certificate Authority (CA) [2]. This may be enhanced by adding a multi-factor authentication process as needed;
2. Maintain Confidentiality: This entails using software with end-to-end unbroken encryption (no in-transit decryption/payload inspection) and Transport Layer Security (TLS) [3];
3. Separate Access and Privilege from Identity: In ZT, this is done by using a Security Token Server (STS). This service uses enterprise policy, and data owner rules together with identity based attributes to establish whether those policy and rules are satisfied. This is in addition to the PKI authentication credential [4];
4. Maintain Integrity – Endpoints use end-to-end TLS integrity measures to confirm that they receive exactly the content that was sent [5];
5. Require Explicit Accountability: ZT approaches log, aggregate, and centrally monitor transactions [6, 7].

A Minimal Instantiation

If the complexity or size of the enterprise dictates an infrastructure need, we can describe the minimal aspects of the necessary infrastructure. The minimal instantiation will provide a core capability that meets the ZT security approach; an access and privilege system that is mostly automated, dynamic, resilient,

secure, and extensible; and an ecosystem that can be enhanced for many of the additional capabilities that are part of the overall ZT architecture.

The following functionalities are required for a minimal instantiation:

1. An attribute store that contains enough information about individual entities so that one can easily ascertain their ability to meet claims requirements.
2. A process for registering enterprise services and the rules and enterprise policies by which claims for access and privilege can be made.
3. A software entity that will generate access claims when a match between the information available for an individual entity in the attribute store matches rules and policies for access and privilege and provide them in the form of credentials that can be validated and verified.
4. A set of user convenience services that allow for corrections and adjustment, make the security requirements user-friendly, and easily maintain the accuracy of the back-office data.

Additionally, at the initial establishment of the Enterprise Attribute Store (EAS), all servers and users are provisioned with PKI certificates. The private keys are stored in Hardware Storage Modules (HSMs) that are kept with the owner. Private keys are never distributed, and only the owner of the private key has access to the use of the private keys stored in the HSM. All servers are configured to require TLS with client certificate authentication and meet strict rules about cipher suites and protocol versions. Communication only takes place if the handshake is a match [8, 9]. All entities and communication paths within the environment are known, so the interfaces, protocols, and authorizations can be strictly controlled.

There are three classes of human entities. Users send browser requests to web applications to request data or services. Administrators conduct similar requests, but they also perform configuration and receive privileged access. Data owners host web applications and services and set the rules and policies for user and administrator access.

There are two types of non-human entities. Web applications and web services provide services and data to requesters according to the rules and policies set by the data owner and the enterprise. Data stores maintain data pertaining to attributes and access rules.

There are four types of interfaces, each with one or more communication types. Legacy interfaces use legacy requests and replies and are secured to the extent possible. Database interfaces are used to access data stores, and they may be full access or read-only depending on the sensitivity of the data and the requesting entity. Browser requests typically use a credentialed authorization, but the authenticated identity may instead be used in some cases when security is strict and requesters are known and limited. Web service interfaces are similar to web application interfaces, but they use web service clients instead of browsers.

Summary

Although there is no specific requirement for infrastructure in a ZT environment, all but the most basic enterprises will require some infrastructure. We have provided the core enterprise environment requirements for a ZT system. This initial build allows for full instantiation of the ZT security model, and the capabilities of an intermediate build for the enterprise environment include an agent-based architecture, access claim delegation, multi-factor authentication, and end-point device management. A larger enterprise may require additional capabilities, including a certificate authority for temporary certificates and active entity veracity measures.

In the DoD, these techniques have been fully developed, tested, and verified on the National Cyber Range and are described in the Air Force Consolidated Enterprise IT Baseline [10-12].

References

[1] Simpson, William R. and Kevin E. Foltz, *Proceedings of The 20th World Multi-Conference on Systemics, Cybernetics and Informatics: WMSCI*, "Enterprise Level Security - Basic Security Model," Volume I, WMSCI 2016, Orlando, Florida, 8-11 March 2016.

[2] X.509 Standards,DoDI 8520.2, *Public Key Infrastructure (PKI) and Public Key (PK) Enabling*, 24 May 2011 and *JTF-GNO CTO 06-02, Tasks for Phase I of PKI Implementation*, 17 January 2006

[3] TLS Family Internet Engineering Task Force (IETF) Standards, RFC 5246 The Transport Layer Security (TLS) Protocol Version 1.2, 2008-08

[4] Organization for the Advancement of Structured Information Standards (OASIS) Open Set of Standards and N. Ragouzis et al., *Security Assertion Markup Language (SAML) V2.0 Technical Overview*, OASIS Committee Draft, March 2008

[5] List, William and Rob Melville, "IFIP Working Group 11.5," *Integrity In Information, Computers and Security*, Volume 13, Issue 4, pp. 295–301, Elsevier, doi:10.1016/0167-4048(94)90018-3, 1994.

[6] Simpson, William R. and Coimbatore Chandersekaran, "An Agent Based Monitoring System for Web Services," CCCT2010, pp. 84–89, Orlando, FL, Apr 2011.

[7] Simpson, William R. and Coimbatore Chandersekaran, "A Multi-Tiered Approach to Enterprise Support Services," 1st International Conference on Design, User Experience, and Usability, part of the 14th International Conference on Human-Computer Interaction (HCII 2011), Orlando, FL, July 2011.

[8] Simpson, William R. and Coimbatore Chandersekaran, "The Case for Bi-lateral End-to-End Strong Authentication," World Wide Web Consortium (W3C) Workshop on Security Models for Device APIs, London, England, December 2008.

[9] Simpson, William R. and Coimbatore Chandersekaran, "Federated Trust Policy Enforcement by Delegated SAML Assertion Pruning," World Wide Web Consortium (W3C) Workshop on Security Models for Device APIs, London, England, December 2008.

[10] Technical Profiles for the Consolidated Enterprise IT Baseline, release 6.0 available at https://intelshare.intelink.gov/sites/afceit/ (CAC required)

[11] Simpson, William R., *Enterprise Level Security – Securing Information Systems in an Uncertain World*, Auerbach Publications, May 2016, ISBN 9781498764452.

[12] Simpson, William R. and Kevin E. Foltz, *Enterprise Level Security 2: Advanced Techniques for Information Technology in an Uncertain World*, Taylor & Francis Group, September 2020, ISBN 9781003080787.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE (DD-MM-YY)	2. REPORT TYPE	3. DATES COVERED (From – To)
00-08-21	Non-Standard	

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Zero Trust Technology Integration Issues	HQ0034-14-D-0001
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBERS

6. AUTHOR(S)	5d. PROJECT NUMBER
Kevin E. Foltz, William R. Simpson	C5223
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESSES	8. PERFORMING ORGANIZATION REPORT NUMBER
Institute for Defense Analyses 4850 Mark Center Drive Alexandria, VA 22311-1882	NS D-22663

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR'S / MONITOR'S ACRONYM
Institute for Defense Analyses 4850 Mark Center Dr., Alexandria, VA 22311	IDA
	11. SPONSOR'S / MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

Project Leader: Kevin E. Foltz

14. ABSTRACT

Zero Trust (ZT) is a popular term within DoD these days. Many products market ZT as a key selling feature, but simply adding a ZT component to an existing non-ZT architecture does not make it ZT. ZT is a philosophy, an approach to security, and an integration of many security techniques. Based on more than a decade of designing, building, and testing a ZT approach for the U.S. Air Force, the sequence of short papers that follow dispel some of the myths and misconceptions that commonly arise with respect to ZT.

15. SUBJECT TERMS

zero trust, single sign-on, segmentation, federation, infrastructure

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Institute for Defense Analyses
Unclassified	Unclassified	Unclassified	Unlimited	25	19b. TELEPHONE NUMBER (Include Area Code)