RESOLVING NETWORK DEFENSE CONFLICTS WITH ZERO TRUST ARCHITECTURES AND OTHER END-TO-END PARADIGMS

William R. Simpson and Kevin E. Foltz

The Institute for Defense Analyses (IDA), Alexandria, Virginia, USA

ABSTRACT

Network defense implies a comprehensive set 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. For most enterprises at this time, that 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. These inspections require decryption of packets and this implies that the defensive suite either impersonates the requestor, or has access to the private cryptographic keysof the servers that are the target of communication. This is in contrast to an end-to-end paradigm where known good entities can communicate directly and no other entity has access to the content unless that content is provided to them. There are many new processes that require end-to-end encrypted communication, including distributed computing, endpoint architectures, and zero trust architectures and enterprise level security. In an end-to-end paradigm, the keys used for authentication, confidentiality, and integrity reside only with the endpoints. This paper examines a formulation that allows unbroken communication, while meeting the inspection and reporting requirements of a network defense. This work is part of a broader security architecture termed Enterprise Level Security (ELS)framework.

KEYWORDS

Appliance, end-to-end security model, ELS, network defenses, web server handlers, zero trust architecture

1. INTRODUCTION

End-to end approaches to network security including distributed computing approaches [1, 2], end-point defenses [3, 4]. Zero Trust Architecture (ZTA) [5, 6], and Enterprise Level Security (ELS) [7, 8].

- Distributed computing is a model in which components of a software system are shared among multiple computers. Even though the components are spread out across multiple computers, they are run as one system. This is done in order to improve efficiency and performance.
- End-point defenses define the requested provider and the requester as the endpoints. Often they use endpoint health indicators and requester identity information to provide fine-grain access control over network resources.
- ZTA uses the principle of protecting individual resources, such as data and computing, within the enterprise instead of protecting the entire internal network at its borders. Requests coming from the internal network are not inherently trusted and must verify their identity and access credentials at each resource. ZTA is designed to prevent data breaches and limit internal lateral movement in the enterprise.

- ELS is a security architecture developed for the US Air Force to overcome the assumptions inherent in fortress defenses.ELS encompasses all of the above methods and is designed from the ground up to be a ZTA. A more complete description of the ELS security architecture is provided in section 3 below.

Entities in the enterprise may be active or passive. Passive entities include storage elements, routers, wireless access points, some firewalls, and other entities that do not themselves initiate or respond to web service or web application requests. Active entities are those entities that request or provide services. Active entities include users, applications, and services.

Although each of these is unique in its security approach, they all share seamless end-to-end encrypted communications in their architectures as shown in Figure 1.Entities in the enterprise may be active or passive. Passive entities include storage elements, routers, wireless access points, some firewalls, and other entities that do not themselves initiate or respond to web service or web application requests. Active entities are those entities that request or provide services according to ELS. Active entities include users, applications, and services. This is a basic conflict with current network defense models, which break this connection at one or more internal components, in effect making them active entities.

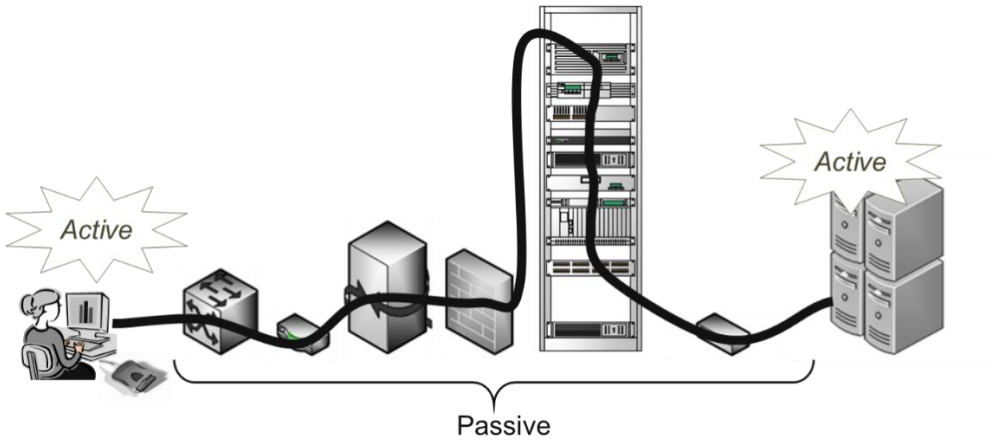

Figure 1. End-to-End Seamless Encrypted Communication

The figure illustrates end-to-end security requires that the front door to the enterprise to be passive, but with the fortress approach has evolved into a very active set of entities. This document follows from our experiences with developing the ELS architecture. ELS proceeds from core tenets to application techniques to core requirements.

2. LITERATURE REVIEW FOR CURRENT DEFENSE PACKAGES

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.” [9] The current defense package assumes that the threat can stopped at the front door, as shown in Figure 2.All traffic in the enterprise, both coming and going are routed through this front door. The front door is often onerous enough that an administrator back door is made available [10] that bypass much of the security check. These backdoors are often the target of exploits.

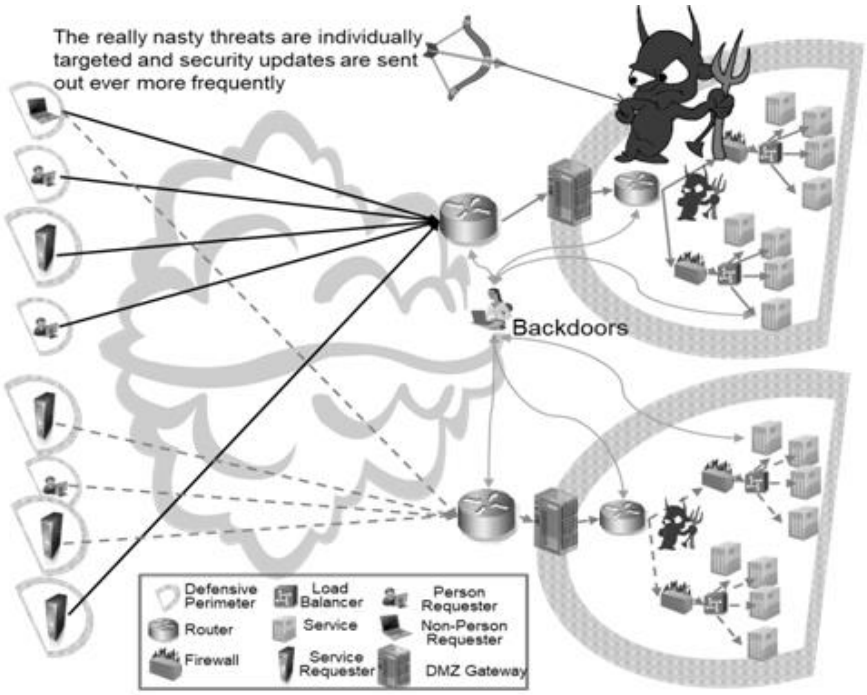

Figure 2. Fortress Protected Enterprises

All active entities in ELS systems have PKI certificates, and their private cyptographic keys are stored in tamper-proof, threat-mitigating storage. Communication between active entities requires bi-lateral, PKI, end-to-end authentication. A verifiable identity claims-based process provides authentication.

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 many operate at line speeds for all communications coming from or going to the enterprise), 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. Figure 3 provides a representation of how these appliances come between the user and the application. The most spectacular result of the network packages shown in Figure 3, is that the fortress defense has spectacularly failed with breaches occurring almost 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. While this is not true at any given time, it is certainly true at various times during operations.

The number and types of appliances can be quite large. Below is a partial list of functional types as provided in the current literature,

- Header-based scanner/logger [11]
  - Views only unencrypted portion of traffic
  - Synchronous or asynchronous operation
  - Scans for defined behavior, logs traffic

- Content-based scanner/logger [12]
  - Views decrypted transport layer security (TLS) content
  - Synchronous or asynchronous operation
  - Scans for defined behavior, and logs traffic/content

- Header-based firewall [13]
  - Views only unencrypted portion of traffic
  - Synchronous operation
  - Scans for and blocks defined behavior

- Content-based firewall – block only [14]
  - Views decrypted TLS content
  - Synchronous operation
  - Scans for defined behavior and blocks (terminates) connection

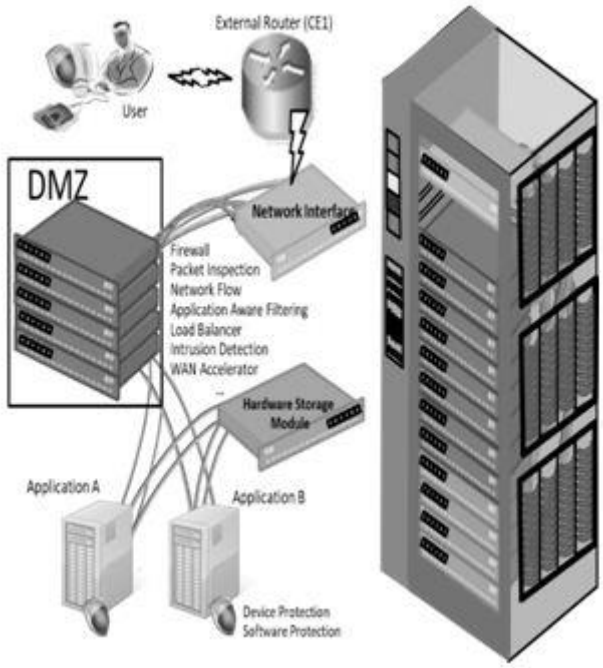

Figure 3. End-Point Access

- Web accelerator [16]
  - Views decrypted TLS content
  - Synchronous operation
  - Modifies content for performance

- WAN accelerator [17]
  - Views decrypted TLS content
  - Multi-party system
  - Synchronous operation
  - Modifies content representation between parties, but no end-to-end modification
- Load Balancers [18]

- Distributes load among destination end-points to improve throughput and reduce latency
- May decrypt content:
  - May combine encrypted flows through a "secure sockets layer (SSL) accelerator"
  - May distribute content by request to different servers based on load
  - These load balancers are active entities
- May not decrypt content:
  - Using "sticky" or end-point balances may route all requests from an entity to the same server
  - These load balancers are passive entities

In order for these appliances to operate at line speeds and process all incoming and outgoing communications, these appliances are built from expensive custom hardware, and they are complex with their own set of vulnerabilities and exploits.

3. HORTCOMINGS OF THE CURRENT APPROACHES

Each of the appliances above offers some functionality and increases the threat exposure. None of these is free from vulnerabilities from a security standpoint, and they do increase the threat surface and the vulnerability space. For example, default passwords or other improperly secured access methods allow an attacker access to any data that the appliance can access. For detailed scans, this could include all decrypted network traffic to and from a server. With a large number of independent appliances, this represents a significant cost and security risk. Use of any appliance must be balanced by the increased functionality and the increased vulnerability. The situation is further complicated by vendor offerings of load balancers with firewall capability, "smart" accelerators that scan content, and software-only offerings that will provide most of these functionalities in a modular fashion.

In this paper, we review the communication models for current network defenses. We then review the inspection processes and its basic architecture. Next, we show how network inspections and reporting are available while maintaining end-to-end communications. Finally, we provide the unique factors that arise with end-to-end approaches and network defenses.

4. THE REAL DE-MILITARIZED ZONE (DMZ)

Figure 4 provides a real-world defense package. Although it may look like a network defense package you have seen, it is not and it is only for illustration purposes. The first thing you see is that it is very complex and has many elements that require proper configuration to function correctly. In reality, it occupies several racks of equipment. Secondly, the first stop after initial entry from the external router is a load balancer that will decrypt the encrypted packets. This is accomplished by either providing the private keys of all servers or allowing the load balancers (LB1 or LB2) to access the hardware storage module (HSM) of the server as if it were the server. While masquerade is generally frowned upon in most security architectures, this is apparently easily allowed in network defense packages. Both break the end-to-end paradigm. Additionally, in most instances, forwarded packets are unencrypted because the appliances are assumed trusted. Each appliance has its own set of vulnerabilities, and this complicates the network defense appreciably.

5. ELS BASIC SECURITY MODEL

The goal of ELS is to provide access to information in an enterprise through secure, validated and verified sharing mechanisms that protect the integrity of the information from creation through utilization. A fully verifiable and validate able process that embodies ZTA principles achieves integrity. ELS is both an architecture and a philosophy that allows intelligent sharing of information among the entities in the enterprise and partners while maintaining a strong security posture that is both uniformly applied and standards-driven throughout the enterprise. ELS is specifically for a high-assurance environment, in which security is of primary importance and attacks on the system are likely to be frequent and sophisticated.

ELS is focused on active entities and their communications. An active entity for ELS is a credentialed requester or provider of information through a web-based interface. This includes human users, non-human requesters, applications, and web services. Active entities have a persistent credential for identity and a temporal credential for access to applications and services. Active entities within the enterprise are registered within the enterprise and have unique identities with associated credentials. Active entities are known identities, and "anonymous" is not one of those identities. Communication between active entities uses identity credentials to perform bi-lateral end-to-end authentication prior to exchanging information. Authorization in the operational environment is implemented by a verifiable short-lived credential with embedded access claims

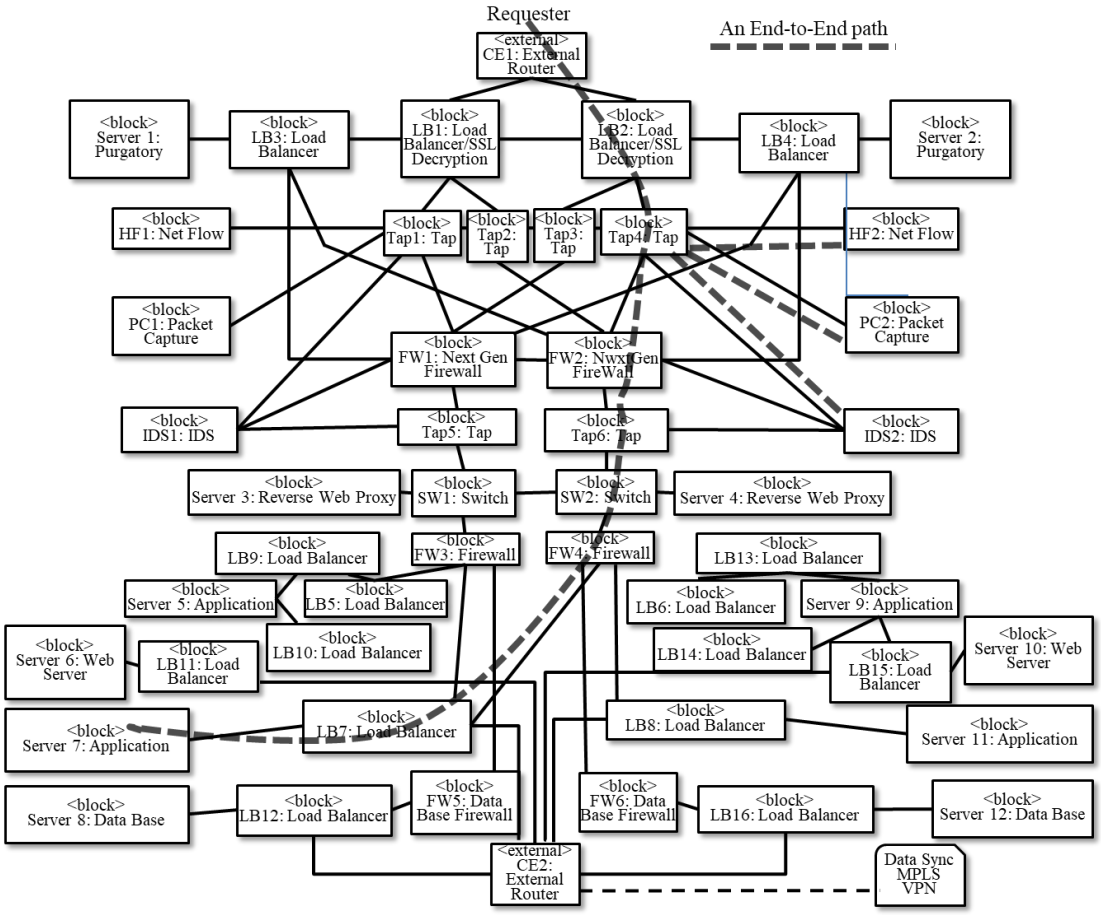

Figure 4 A Real DMZ

Claims represent satisfaction of access control rules and are included as part of an authorization credential issued and signed by a trusted credential issuer. The access control rules for a data set are provided by the data set's owner. The data owner may also request, as part of the access control requirement definition, additional information about the requesting entity to determine the level of privilege.

5.1. System Design and Maintenance

For system design and maintenance, a set of core tenets is the starting point. These describe the desired highest-level properties and design philosophies of the system. They do not indicate what to build but provide guidance that influences all decisions about what to build, how to build it, and the choices of finer grained details. From these tenets, key concepts describing the system to be built are derived. The concepts describe what to build at the highest level. They are not sufficient to build the system, but they provide a vision of some of the critical parts and how they interact. From these concepts, a set of requirements are developed, which are closely tied to the concepts and provide sufficient detail to start building the system. The idea is that an enterprise can use these requirements as the foundation for building a system and supplement them with additional details as the design is refined.

This method bridges the gap between the builder of a system, who is focused on implementation details, and the designers of the architecture, who focus on the high-level properties of the system. It also enables a systematic assessment by tying requirements to the overall design goals of the system. This facilitates modification to the system by showing which tenets, concepts, and requirements are affected when one or more of them change due to changes in technology or adjustments to goals.

The idea of the basic security model can be visualized in Figure 5. The tenets are like solid, heavy rocks that are positioned in the beginning and form the structure for everything else to build upon. These rarely change, and when they do it reflects a major change in direction or external circumstances.

The concepts are represented as wood, which is solid like rock, and can last a long time, but the structures they build require maintenance and repair. For wooden structures, components can break or rot, but with maintenance and repair they can last a long time. Concepts are meant to last and be structural elements, but they are not as solid or resilient as the tenets. The concepts are tailored to the system under construction, and they are easier to change than the tenets. Just as the particular system is more likely to change than the overall goals, concepts are more likely to change than tenets.

The requirements are generally derived from the concepts, and paper, a wood derivative, is used to represent them in the next layer. The requirements are more flexible and represent the particular choices for the system functions, which may change more rapidly than the functions themselves. Paper, being easily folded, shredded, moved, and otherwise changed, represents the idea that requirements may change more often than concepts.

The linkages are not shown, but they are an important part of the model, as they define the structural connections from the tenets to concepts and requirements, much like an architectural diagram can show relationships between a rock foundation, a wooden external structure, and paper elements within.

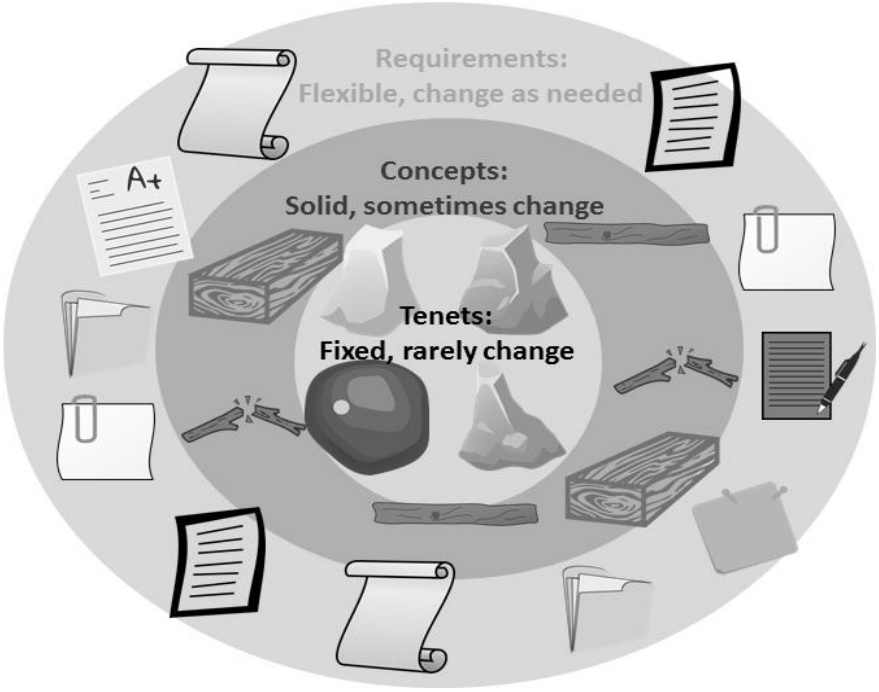

Figure 5 Basic Security Model Visualization

This bullseye representation is the center of the system model. Additional rings can be layered outside of this core, including the following:

- Additional detailed functional requirements
- Implementation, including products, their versions, and configurations
- Operational guidance on how to use the products

By continuing the linkages outward a mapping can be made from tenets to operational guidance through the intermediate layers. Changes to the internal layers, especially tenets and concepts, have a large effect, as such changes generally propagate outward along their connections. The goal of this design method is to design the architecture to address changes at the outermost level possible for the type of change it is. Major changes will necessarily be addressed near the center and have significant effects on system design, but small changes in a properly designed system should only affect the fringes and result in quick fixes using standard methods.

The sections below describe the tenets, concepts, and requirements in more detail. Examples are provided for the development of the ELS system.

5.2. Core Tenets

The definitive document of zero trust architectures is NIST 800-207 [19]. This document begins with six core tenets. ELS started with 16 core tenets. The tenets are the drivers of all architectural decisions. A complete list of the core tenets are provided in [7, 8]. Some of the ELS tenets are as follows:

0. **The enemy is present**. Malicious entities are present and our systems need to function with these embedded threats rather than rely on filtering them out.

1. **Simplicity**. Added features come at the cost of greater complexity, less understandability, greater difficulty in administration, higher cost, and/or lower adoption rates that may be unacceptable to the organization.
2. **Extensibility**. Any construct should be extensible to the domain and the enterprise, and ultimately across the enterprise and coalition.
3. **Information hiding**. This involves revealing to the requester and the outside world only the minimum set of information needed for making effective, authorized use of a capability.
4. **Accountability**. This means being able to unambiguously identify and track which active entity in the enterprise performed each operation.

The tenets generally fall into two categories: must-haves and design principles. Tenets 0 and 4 are examples of must-haves. ELS must be able to function with malicious entities that are attacking from outside and inside the system, and it must provide accountability. Simplicity, extensibility, and information hiding are examples of design principles. These are not must-haves, as they are always in some tension with each other. It is not the absolute value of these tenets that matters but the relative values and their balance. For example, a complex solution may be acceptable if the goal itself is inherently complex. Simplicity means that the complexity in the system reflects the complexity of the goal.

The tenets for most projects will be similar. There may be differences, such as valuing anonymity over accountability in a privacy-based system, but things like simplicity and extensibility are common design themes that are likely to be repeated broadly beyond just enterprise security.

5.3. Key Concepts

The key concepts are based on the tenets and address specific architectural decisions that relate to the requirements. These are likely to be similar to the ELS concepts for many security-based systems, but different for projects with other goals. The concepts form a bridge between the high-level tenets and the technical requirements by describing the high level system in a way that maps to the tenets. The complete list of ELS Key Concepts is provided in [7. 8]. A subset of the ELS concepts follows:

0. ELS-specific concepts. These are important choices based on current technology. Due to their overall importance to the ELS architecture, they are considered as a single concept.
    a. PKI credentials are used for active entity credentials.
    b. Security Assertion Markup Language (SAML) with claims is used for authorization credentials.
    c. TLS v1.2 is used for end-to-end confidentiality, integrity, and authentication.
    d. A Security Token Server (STS) is the trusted entity for generating authorization credentials.
    e. Exceptions in an implementation of an ELS system must have a documented plan and schedule for becoming compliant.
1. A standard naming process is applied to all active entities.
2. Authentication is implemented by a verifiable identity claims-based process.
3. Identity claims are tied to a strong vetting process to establish identity.
4. Active entities verify each other's identity.
5. The verification of identity is by proof of ownership of the private key associated with an identity claim.
6. Active entities act on their own behalf.

Concepts are linked to the tenets. Linkages are shown in Figure 6. The connections between tenets and concepts are important for future changes as they allow traceability and a way to determine the effects of changing any of the tenets or concepts on associated concepts or tenets.

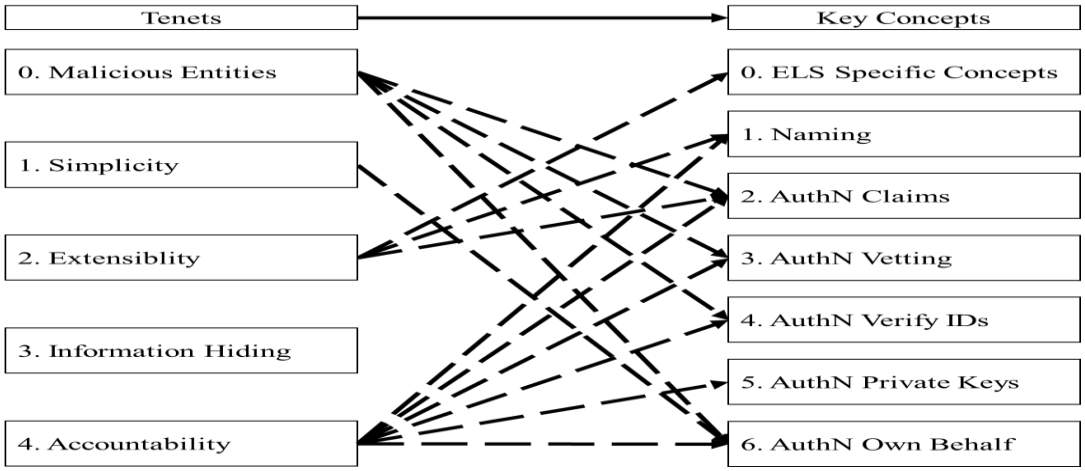

Figure 6 Mapping ELS tenets to concepts

The ELS-specific concepts are a collection of important protocols and standards that are to be used across the enterprise. Although each of these could be taken individually as a requirement, they together form such an important part of the ELS model that they are elevated to the level of a concept. The other concepts listed all relate to authentication, which is an important part of the ELS model.

5.4. Technical Requirements

The technical requirements are based on the key concepts and are traceable through the concepts to the core tenets. A complete list of the derived requirements is given in [7, 8]. A subset of the requirements for ELS follows:

1. Active entities shall be named in accordance with a Naming standard.
2. Active entities within the enterprise shall have unique identities.
3. Active entities shall use credentials from approved certificate-issuing authorities.
4. Active entity communication shall use two-way, end-to-end PKI authentication.
5. No active entity shall be anonymous.
6. Authentication tokens shall not be allowed.
7. Traditional single sign-on shall not be allowed.
8. Private keys shall be stored in tamperproof, threat-mitigating storage to which only the associated entity has access.
9. Impersonation of active entities through sharing of private keys or issuing of duplicate credentials shall not be allowed.
10. Proxies or portals shall not be allowed, because they cause ambiguity in identity.
11. Active entity authentication shall use only primary or derived credentials

The concepts and requirements are generally more closely related than the tenets and concepts. The authentication related requirements generally reference the authentication related concepts, whereas the tenets connect more uniformly across the concepts.

The connections between concepts and requirements are shown in Figure 7. Because these concepts and requirements are all related to authentication, there are many links between them. By combining the tenet to concept and concept to requirement connections the paths between tenets and requirements can be shown. ZTA is at the core of ELS in that 21 of the derived requirements relate to ZTA. The full mapping of all ELS tenets, concepts, and requirements is shown in [7, 8].

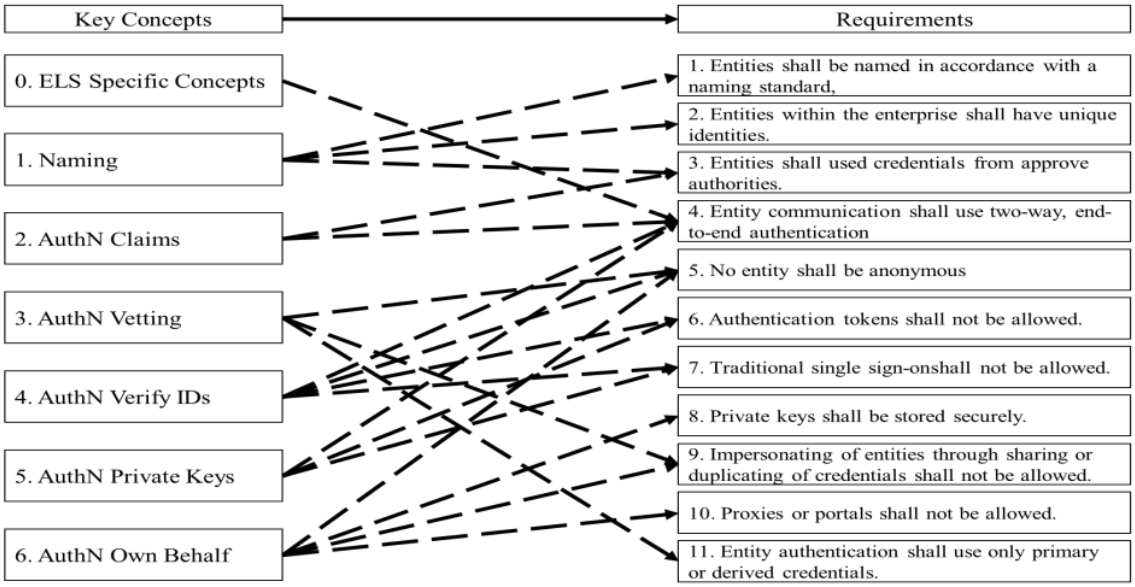

Figure 7 Mapping concepts to requirements

5.5. Mappings Lead to Derived Requirements

The full mapping can be used to trace requirements back to concepts and tenets, which can help in making and justifying implementation decisions. For example, the enterprise may consider inserting a proxy in front of a server and sharing the server's certificate and private key with the proxy to enable in-depth security scans on incoming encrypted traffic. This is a common practice, but it results in the following ELS violations:

- Requirement #2 – the proxy shares the same name as the server by using its certificate and private key.
- Requirement#4 – the proxy breaks the end-to-end authentication by acting as the server.
- Requirement#8 – the proxy is not the appropriate entity to access the server's private key.
- Requirement#9 – the proxy impersonates the server.
- Requirement#10 – the proxy causes ambiguity in the server's identity.
- Requirement#12 – the proxy has no claims but is accessing the server.
- Requirement#14 – the proxy has no attributes.
- Requirement#22 – the proxy breaks the end-to-end TLS connection.

Tracing these requirements back to related concepts, we see that the most often referenced is Concept 6, "Active entities act on their own behalf." The proxy is a direct violation, since it acts on behalf of the server when communicating with requesters. Others with multiple references are Concept 5, "The verification of identity is by proof of ownership of the private key associated with an identity claim," which again is violated directly by sharing the private key of the server with the proxy. Also Concept 8, "Service providers use identity and authorization credential

claims to determine access and privilege," which is violated because the proxy gains access to the service without valid identity or authorization credentials.

Extending this process, we can link back from these concepts to the related tenets. The most referenced are Tenet 0, "Malicious entities are present," Tenet 4, "Accountability," Tenet 2, "Extensibility," and Tenet 11, "Trust but verify." When using proxies we provide more points of exposure to enemies, we reduce accountability by spreading identities across multiple nodes, and we reduce the ability to verify and validate identity. Extensibility is affected less directly, but many of the choices made for extensibility are negated by using proxies.

The example of proxies was chosen to illustrate a serious violation. Other changes might have minimal impact. For example, choosing not to scan outputs for consistency would violate Requirement 27, which maps only to Concept 21, and Tenets 0 and 15.

5.6. Benefits

The benefits of using tenets, concepts, and requirements to guide the development process depend on the goal of the system to be built. A general benefit is the continued adherence to initial design goals throughout all the decisions in the development process.

For ELS the benefits can be grouped into three major categories, as illustrated in Figure 8. The first is security. Security is the main design goal for ELS, and adhering to the original tenets through all the changes and decisions in the design process helped to maintain a strict adherence to this goal despite constant outside influences that attempted to impose their own goals at the expense of security.

A second benefit is cost savings. By designing the system to address changes at the fringes of Figure 5, less time is spent redesigning the system, since changes are smaller and can be easily addressed by established procedures. In contrast, redesigning the architecture every time a product or component is swapped out requires a large level of effort. This is often the case when there is no forethought in designing a system.

A third benefit of using this model is dealing with vendors. This basic model provides an architecture for the system into which vendor products can fit. The alternative is to adjust the architecture to fit available vendor product suites. Vendors will often sell a product that is a collection of smaller pieces, and then slowly add more pieces in an effort to integrate all functions under their product suite. This is convenient and efficient in many cases, but it locks the system architecture to a particular vendor and product, which can cause problems when enterprise needs and vendor product functionality diverge. The explicit mapping of the basic security model and choice of widely used protocols and standards maintains a focus on functions instead of products.

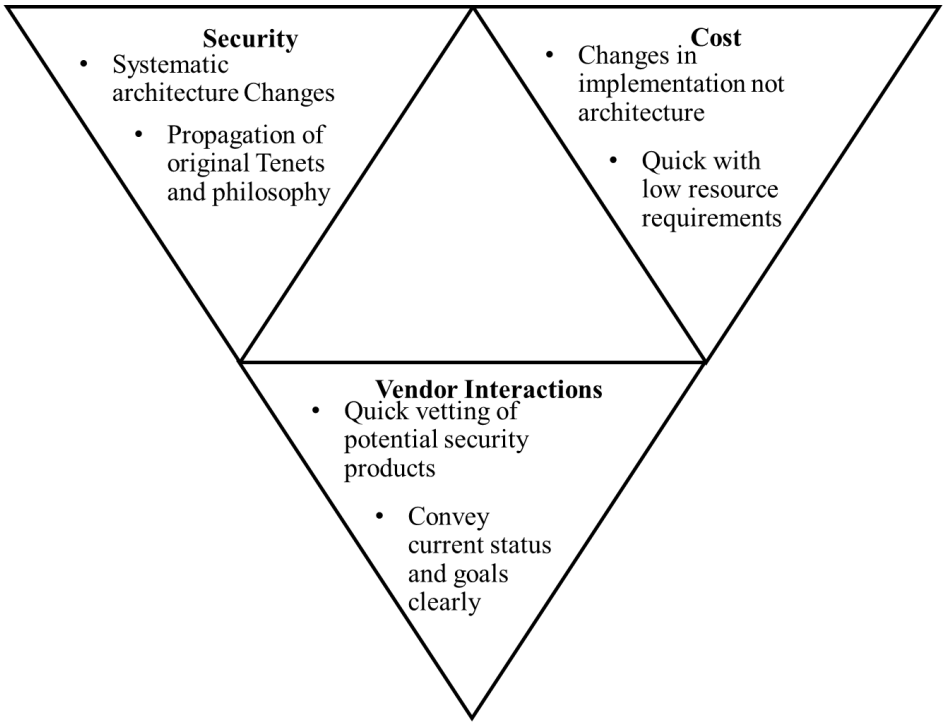

Figure 8 Benefits of using the model

6. THE ELS FRAMEWORK

The ELS framework has evolved from a fortress approach, in which the assumption that the threat is stopped at the front door, to a distributed security system that eliminates or mitigates many of the primary vulnerability points inherent with that system, as shown in Figure 9.The basic process of identification involves a two-way contract between two entities that are initiating a communication. Each entity needs to have some assurance that the entity they are engaged with is known and, specifically, the entity to whom the communication should be allowed. The presentation of claims by each entityis verifiable and validateable. These claims are often in the form of credentials. [20, 21] provide an extensive description of these processes.

Entities may be active or passive. Passive entities include storage elements, routers, wireless access points, some firewalls, and other entities that do not themselves initiate or respond to web service or web application requests. Active entities are those entities that request or provide services according to ELS. Active entities include users, applications, and services. All active entities have PKI certificates, and their private keys are stored in tamper-proof, threat-mitigating storage. Communication between active entities requires bi-lateral, PKI, end-to-end authentication. A verifiable identity claims-based process provides authentication.

The real problem is now manifest because the active inspection content and reporting of information are not in the original list of derived requirements, and if the seamless encrypted end-to-end communication is broken, the basic concept of integrity and ZTA are broken.

7. A DISTRIBUTED APPROACH TO COMPUTING

Many of the new approaches to security have moved to a distributed security approach. The ELS framework has evolved from a fortress approach, in which the assumption that the threat is

stopped at the front door, to a distributed security system that eliminates or mitigates many of the primary vulnerability points inherent with that system, as shown in Figure 9. The basic process of identification involves a two-way contract between two entities that are initiating a communication. Each entity needs to have some assurance that the entity they are engaged with is a known entity and, specifically, the one to whom the communication should be allowed. The presentation of claims by each entityis verifiable and validateable. These claims are often in the form of credentials. [7, 8] provide an extensive description of these processes. 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.

8. A SOLUTION TO THE CONFLICT– CREATING THE PSEUDO-APPLIANCE

The main contribution of the ELS approach to network defense in a distributed system is in maintaining the inspection process without breaking the end-to-end encryption of communications. The pseudo-appliance captures all of the inspection processes and places them into one software process that resides in the application. This is the first step in realigning the priorities between the current approach and the end-to-end approach, as shown in Figure 10.The path from the user to the application in the top half of the figure shows the processes needed for inspection. Note that the private key for server 7 has been hand passed to the initial load balancer so that the exchange of information is visible. Next, the load balancer decrypts packets for inspection. This includes not only the inspection, but also the necessary reporting.

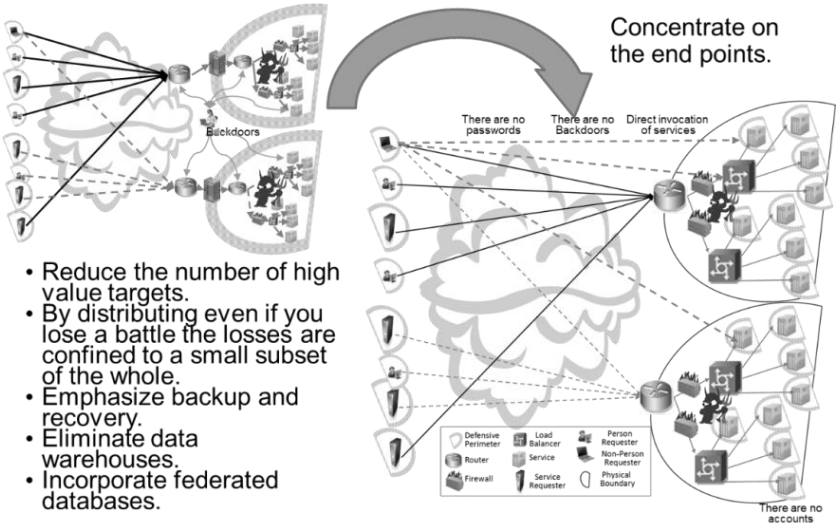

Figure 9 ELS Distributed Security Architecture

In the second half of Figure 10, we show the user directly communicating with the load balancer in front of the application (which now contains the inspection process).We have reduced the bandwidth necessary to handle the traffic at the network interface and distributed the computing burden. Tagging the communications between the requester and provider bypasses the DMZ stack. The initial handshake (which is unencrypted) includes the exchange of two white-listed PKI certificates. This exchange in ELS is the bi-lateral authentication of entities and is the initial setup for TLS encryption of all communications. This exchange allows for this tagging. As the decryption now occurs in server 7 prior to inspection, key passing is no longer required, and the end-to-end confidentiality is maintained. Untagged traffic will go through the normal DMZ processing. The reduction in traffic bandwidth at the front door may reduce the need for more expensive in-line processors and several of the downstream load balancers.

Figure 11 shows the handler makeup in the server. The handlers rely upon software only versions of the inspection systems. Software only versions are not always not available. To counter the problem of a potential threat identification, an interface in the handler is provided that allows the handler to sends packets or contents to a hardware appliance version of the inspection for completeness purposes. The application, itself, may await this inspection or proceed with processing, a=depending upon configuration.ELS enhances protection of the application server and provides additional security protections as discussed in the following section.

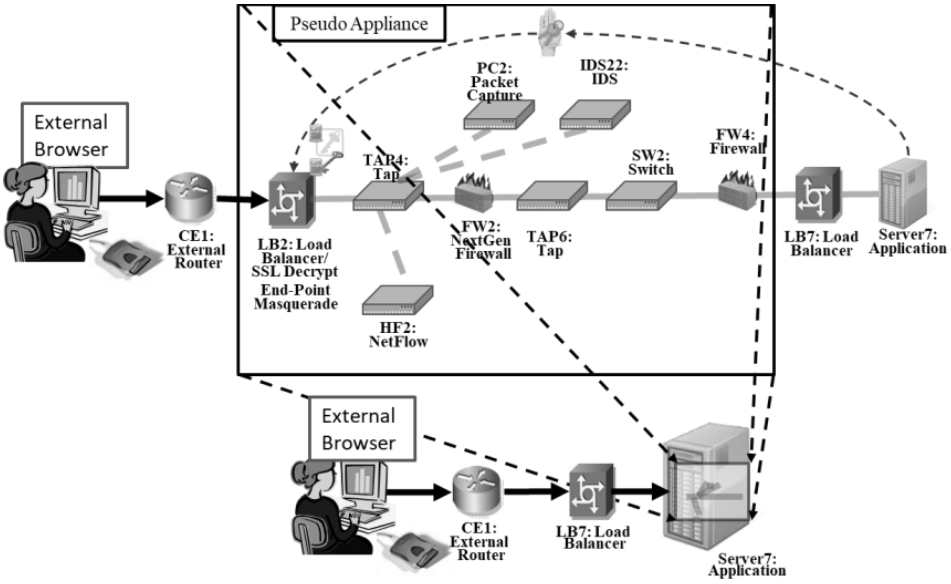

Figure 10 Creating the Pseudo Appliance

9. END-POINT PROTECTION SYSTEMS

The end-point protection system must provide firewall functionality under certain circumstances (as shown in Figure 12) based on end-point, claimed identity, requested action, and other factors. This protection system is in addition to the end-point pseudo appliance, and forms a more complete security architecture.

- Black list – The only functionality enforced is block or drop packets. The black list is centralized, managed, and "pushed" to the protection system (ELS compliant)
- White – Varying degree of firewall enforcement based upon device and criticality. White membership includes The Security Token Server (STS), for example.
- Gray – Full firewall functionality is enforced. Functionality includes virus scan, malware scan, and other deep packet techniques.

The protection system has the capability to monitor, filter, or shut down traffic to given ports. It scans for malicious code. It examines devices for geo-location, veracity, system login and other elements. It examines incoming and outgoing traffic for anomalies or known exploits. It acts in the security context of the end-point for both requester and provider and examines not only the encrypted traffic but also the clear text traffic for malicious behavior or code. This requires access to the unencrypted traffic as well as the encrypted traffic. The protection system provides most but not all of the checks. Figure 12 walks through checks in an ELS enclave provided by the protection system, the server handlers, the service handlers, and the service itself, minimizing the need for in-line appliances.

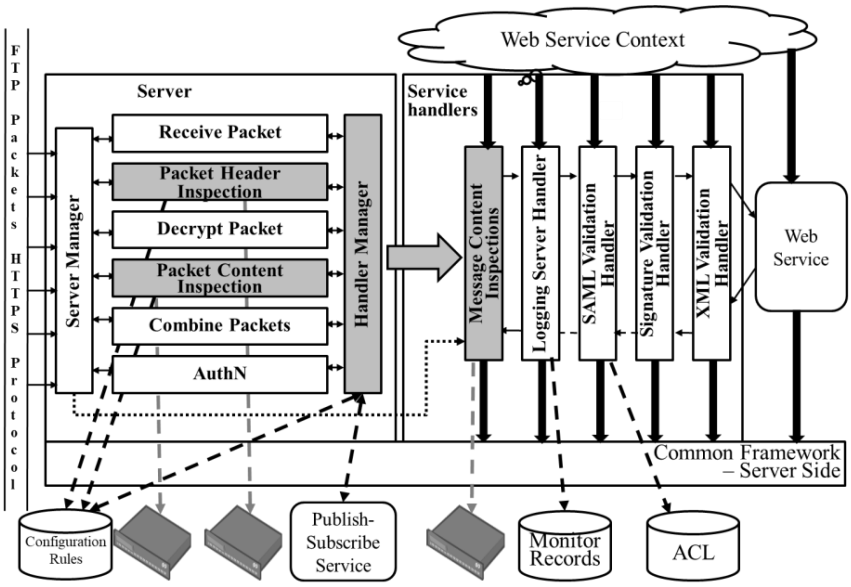

Figure 11 ELS End-point Network Security Functions

This capability of the protection system is defined in terms of functional elements, some of which are listed below:

- Identifying unsafe websites during searches
- Detecting and repairing computer problems
- Enforcing policies on local machine
- Monitoring asset configurations and compare against baseline to detect changes
- Preventing use of unauthorized USB and flash media
- Blocking known and unknown buffer overflow exploits
- Preventing malicious code installation/execution
- Identifying activities that deviate from organizational policy
- Ensuring firewall functionality
- Monitoring DHCP requests on the network
- Marking any system that does not check in as rogue
  - ⇒ Includes device veracity checks
  - ⇒ Includes geo-location limits
- Scanning for compliance with policies
- Identifying host vulnerabilities on the network
- Making data available to the consumer, using ELS security
- Providing situational awareness
- And others as indicated by threat modelling

The end-point protection system maintains an inventory of what is present (virtual and real) on all devices in the enterprise. Regular updates to this list ensures timely measures can be taken when an incident occurs. The protection system scans applications, configurations, permissions, services, registry entries, and other attributes to ensure that any changes from the baseline configuration have proper authorization. Any unauthorized or questionable differences from an approved baseline are reported to a central monitoring facility.

The protection system detects and removes malicious software from email by extracting, sandboxing and executing attachments to email in the user's security context before the user can

do this. The execution is monitored and if malicious the attachment is removed from the email and forwarded to the security team for further analysis. Phishing can overcome people's mistrust of such attachments; this is an important part of device protection.

To prevent web-based attacks, the protection system flags potentially malicious sites to warn users. The protection system uses both heuristics and historical data to determine whether a site is safe or not. As search accesses many new sites, this is the ideal time for performance of such protection functions.

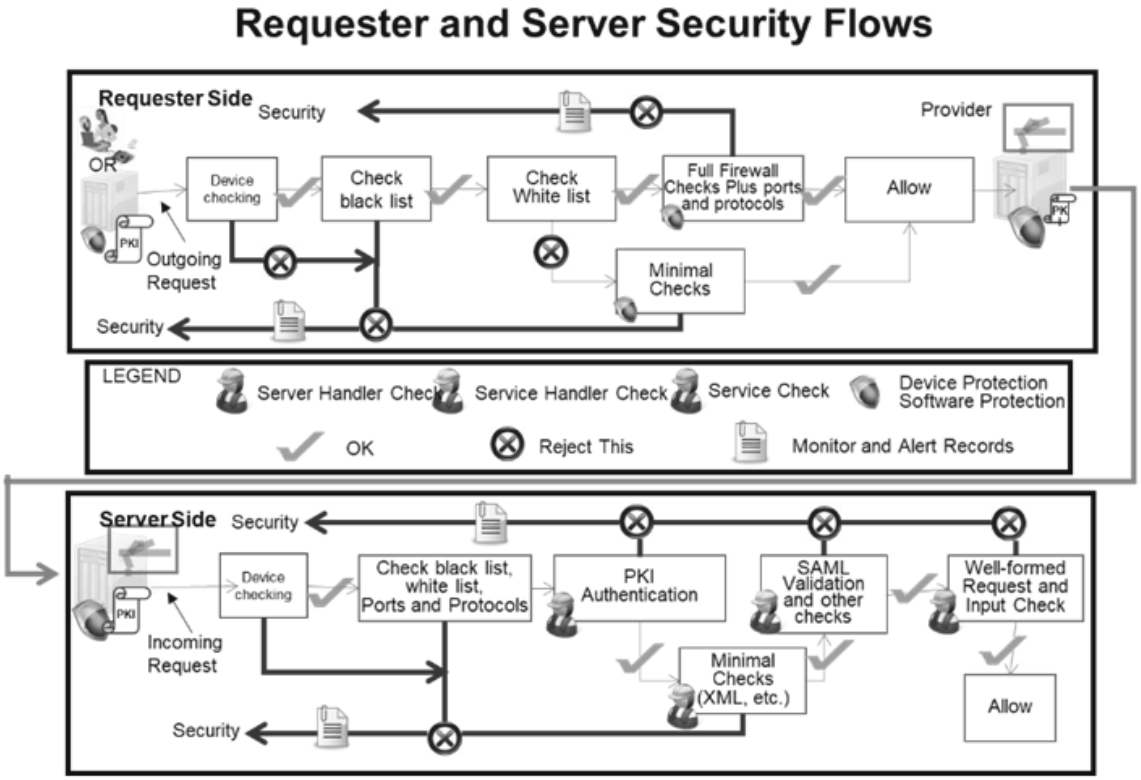

Figure 12 Protection Provided Without In-Line Appliances

The protection system provides mechanisms to fix problems. Of course, a fully compromised system might be unresponsive to commands to fix certain issues, so this is not always possible. However, for most problems, fixing the problem remotely instead of requiring on-site manual accessis the best course of action.

The protection system enforces policy on the local machine and enforcement of group policy or other methods for setting policy for compliance. Policies that are not enforced by the device itself must be monitored explicitly by the protection system.

The protection system keeps an accurate record of what the approved baseline configuration is for a given device [22]. After a scan of the device, any differences are recorded and made available to the central monitor.

With new threats evolving through non-standard interfaces, such as USB, printers, and other attached devices, the protection system provides a way to manage these interfaces, either by monitoring or filtering traffic on them, disabling them, or using other methods to prevent attacks from these sources.

By closely monitoring code execution, the protection system prevents buffer overflows. Low-level system calls are monitored to track any attempts at writing to unallocated memory spaces, stopping both known and unknown buffer overflows from being exploited. This type of monitoring and prevention requires elevated privilege, because requiresaccess to system level resources, not just user data.

The protection system stops a user from installing new executable code, unless they are explicitly authorized. This prevents a user from compiling and running code downloaded from, or modified by, a malicious entity. It also provides a generic catch-all for any executables that may have bypassed the email or web monitoring functions. By stopping the user from installing executables, the protection system also stops malicious entities from using hijacked user accounts or sessions to run malicious code.

Enterprise enforcement of rules that govern behavior on their networks and devices is partially achieved by the protection system [22].Although many of these rules will already be handled through group policy or device Security Technical Implementation Guides (STIG), some activity can only be monitored dynamically through the protection system. For example, use of TLS with appropriate version, ciphers, two-way authentication using PKI, and use of appropriate extensions are not typically monitored with existing tools in most commercial enterprises and must be implemented by the protection system.

10.CONCLUSIONS

We have reviewed the ELS security model and the end-to-end requirement within the enterprise. We have also reviewed “normal” network defense processes and described the issues that the current network defenses raise and the vulnerabilities introduced. Finally, we have provided an end-to-end approach that allows for both network inspection and reporting and the maintaining of unbroken encryption to the final destination, including enhanced defensive protections afforded by ELS. This approach identifies the instances of official business and deferring the initial inspection until arrival at the target server. For enterprise operations, defining a clear end-to-end approach means a reduced attack space. The approach also reduces bandwidth requirements at the front door of the enterprise and may reduce the need for some load balancing. We have also reviewed the specific requirements for an enterprise level security that is bi-laterally authenticated and encrypted end-to-end. This is part of a body of work for high-assurance enterprise computing using web services. Elements of this work are described in [23-34].

REFERENCES

[1] Tanenbaum, Andrew S.; Steen, Maarten van (2002). Distributed systems: principles and paradigms. Upper Saddle River, NJ: Pearson Prentice Hall. ISBN 0-13-088893-1.

[2] Magnoni, L. (2015). "Modern Messaging for Distributed Sytems (sic)". Journal of Physics: Conference Series. 608 (1): 012038. doi:10.1088/1742-6596/608/1/012038. ISSN 1742-6596.

[3] Hassan N.A. (2019) Endpoint Defense Strategies. In: Ransomware Revealed. Apress, Berkeley, CA. https://doi.org/10.1007/978-1-4842-4255-1_4

[4] Mark Khai Shean Tan, Sigi Goode & Alex Richardson (2020) Understanding negotiated anti-malware interruption effects on user decision quality in endpoint security, Behaviour & Information Technology, DOI: 10.1080/0144929X.2020.1734087

[5] Dayna Eidle, Si Ya Ni, Casimer DeCusatis, Anthony Sager, "Autonomic security for zero trust networks", Ubiquitous Computing Electronics and Mobile Communication Conference (UEMCON) 2017 IEEE 8th Annual, pp. 288-293, 2017.

[6] C. DeCusatis, P. Liengtiraphan, A. Sager and M. Pinelli, "Implementing Zero Trust Cloud Networks with Transport Access Control and First Packet Authentication," 2016 IEEE International Conference on Smart Cloud (SmartCloud), New York, NY, 2016, pp. 5-10, doi: 10.1109/SmartCloud.2016.22.

[7] Simpson, William R., CRC Press, "Enterprise Level Security – Securing Information Systems in an Uncertain World",by Auerbach Publications, ISBN 9781498764452, May 2016, 397 pp.

[8] Simpson, William R. ank Kevin E. Foltz,CRC Press, "Enterprise Level Security 2: Advanced Techniques for Information Technology in an Uncertain World," by Taylor & Francis Group, September 2020, 338 pp, ISBN 9781003080787.

[9] Science Direct, Editors: Jason Andress, Steve Winterfeld, Cyber Warfare, ISBN 9781597496377, 2011, Jason Andress, Steve Winterfeld, Chapter 10 - Computer Network Defense, Pages 179-191, http://www.sciencedirect.com/science/article/pii/B9781597496377000101, last accessedon 24 Novemeber 2020.

[10] TechTarget.com. "backdoor (computing)," https://searchsecurity.techtarget.com/definition/back-door, last accessed 22 November 2019.

[11] Jack Wallen, "Five free, dead-easy IP traffic monitoring tools," Tech Republic, September 2011, https://www.techrepublic.com/blog/five-apps/five-free-dead-easy-ip-traffic-monitoring-tools/, last accessed 22 November 2019.

[12] Moskovitch R, Elovici Y. "Unknown malicious code detection – practical issues.", In Proceedings of the 7th European Conference on Warfare and Security (ECIW'08), Plymouth, UK, 2008.

[13] A. Begel, S. McCanne and S. L. Graham, BPF+: Exploiting global data-flow optimization in a generalized packet filter architecture, in: Proc. of ACM SIGCOMM, Cambridge, MA, USA (1999) pp. 123–134.

[14] M. McDaniel and M.H. Heydari, "Content Based File Type Detection Algorithms," Proceedings of the 36th Annual Hawaii International Conference on System Sciences, IEEE, ISBN: 0-7695-1874-5, DOI:10.1109/HICSS.2003.1174905, Jan 2003.

[15] Mike Fisk and George Varghese, "Fast Content-Based Packet Handling for Intrusion Detection," Los Alamos National Lab Computing Communications and Networking Division, May 2001, https://apps.dtic.mil/dtic/tr/fulltext/u2/a406413.pdf, last accessed 22 November 2019.

[16] Jian Song and Yanchun Zhang. 2007, "Architecture of a Web Accelerator for Wireless Networks", In Proceedings of the thirtieth Australasian conference on Computer science - Volume 62 (ACSC '07), Gillian Dobbie (Ed.), Vol. 62. Australian Computer Society, Inc., Darlinghurst, Australia, Australia, 125-129.

[17] Shin-ichi Kuribayashi, "Improving Quality of Service and Reducing Power Consumption with WAN Accelerator in Cloud Computing Environments," International Journal of Computer Networks & Communications (IJCNC) Vol.5, No.1, January 2013.

[18] Afzal, S., Kavitha, G. "Load balancing in cloud computing – A hierarchical taxonomical classification." J Cloud Comp 8, 22. Decemeber 23, 2019, https://journalofcloudcomputing.springeropen.com/articles/10.1186/s13677-019-0146-7

[19] Scott Rose, Oliver Borchert, Stu Mitchell, and Sean Connelly, National Institute of Standards and Technology, NIST Special Publication 800-207, Zero Trust Architecture, August 2020,https://doi.org/10.6028/NIST.SP.800-207 , last accessed on 24 November 2020.

[20] PKI Standards: PKCS#12 format PKCS #12 v1.0: Personal Information Exchange Syntax Standard, RSA Laboratories, June 1999 http://www.rsa.com/rsalabs/node.asp?id=2138 PKCS 12 Technical Corrigendum 1, RSA laboratories, February 2000.

[21] Organization for the Advancement of Structured Information Standards (OASIS) open set of Standards S. Cantor et al.,"Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0," OASIS Standard, March 2005

[22] William R. Simpson and Kevin E. Foltz, Lecture Notes in Engineering and Computer Science, Proceedings of the World Congress on Engineering (WCE) 2018, "Enterprise End-point Device Management", pp. 331-336, Imperial College, London, 4-6 July 2018, IBSN: 978-988-14047-9-4, ISSN: 2078-0958.

[23] William R. Simpson and Kevin E. Foltz, Lecture Notes in Engineering and Computer Science, Proceedings World Congress on Engineering and Computer Science(WCECS) 2017, Volume 1, "Enterprise Level Security: Insider Threat Counter-Claims", pp112-117, Berkeley, CA. October 2017.

[24] William R. Simpson and Kevin E. Foltz, Proceedings of the Information Security Solutions Europe (ISSE) 2016, ISBN:9781541211445,"The Virtual Application Data Center", pp. 43-59, https://www.amazon.com/isse2016-3-Information-Security-Solutions-Europe/dp/1541211448, Paris, France, November 2016.
[25] William R. Simpson and Kevin E. Foltz,Haeng Kon Kim • Mahyar A. Amouzegar (eds.), Transactions on Engineering Technologies, Special Issue of the World Congress on Engineering 2015, Chapter 15, pp. 205-220, "High Assurance Asynchronous Messaging Methods", 15 pp., DOI 10.1007/978-981-10-2717-8, Springer Dordrecht 2017.
[26] William R. Simpson and Kevin E. Foltz, Lecture Notes in Engineering and Computer Science, Proceedings of the World Congress on Engineering (WCE) 2017, "Assured Identity for Enterprise Level Security", pp. 440-445, Imperial College, London, July 2017, IBSN: 978-988-14047-4-9.
[27] William R. Simpson and Kevin E. Foltz, Proceedings of The 21th World Multi-Conference on Systemics, Cybernetics and Informatics: WMSCI, "Data Mediation with Enterprise Level Security",WMSCI 2017, Orlando, Florida, 8-11 July 2017, 6 pages.
[28] William R. Simpson and Kevin E. Foltz, Proceedings of the 22nd International Command and Control Research and Technology Symposium (ICCRTS), "Escalation of Access and Privilege with Enterprise Level Security", ISBN: 978-0-9997246-0-6, Los Angeles, CA. September 2017.
[29] William R. Simpson and Kevin E. Foltz, Sio-Long Ao, et. al. (eds.), IAENG Transactions on Engineering Sciences, Special Issue of the Association of Engineers Conferences 2016, Volume II, pp. 475-488, "Electronic Record Key Management for Digital Rights Management", 14 pp., World Scientific Publishing, Singapore, ISBN 978-981-3230-76-7, 2018.
[30] William R. Simpson and Kevin E. Foltz, "Secure Identity for Enterprises," IAENG International Journal of Computer Science, vol. 45, no. 1, pp 142-152, ISSN: 1819-656X, February 2018.
[31] William R. Simpson and Kevin E. Foltz, Proceedings of the 8th International Conference on Electronics, Communications and Networks (CECNet 2018), Volume 1, "Cloud Security and Scalability", pp 27, Bangkok, Thailand, November 2018.
[32] William R. Simpson and Kevin E. Foltz, "Insider Threat Metrics in Enterprise Level security," IAENG International Journal of Computer Science, vol. 45, no. 4, pp 610-622, ISSN: 1819-656X, December 2018.
[33] Simpson W. and Foltz K., Lecture Notes in Engineering and Computer Science, Proceedings World Congress on Engineering and Computer Science 2015, Volume 1, "Maintaining High Assurance in Asynchronous Messaging," pp. 178–183, Berkeley, CA, October 2015.
[34] William R Simpson, and Kevin E. Foltz, "Mobile Ad-hoc for Enterprise Level Security," Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering and Computer Science 2018, 23-25 October, 2018, San Francisco, USA, pp172-177.

AUTHORS

Dr. Simpson has over two decades of experience working to improve systems security. He has degrees in Aeronautical Engineering and Business Administration. He also attended several schools for military and government training. He spent many years as an expert in aeronautics before delving into the field of electronic and system test, and he has spent the last 20years on IT-related themes (mostly security, including processes, damage assessments of cyber intrusions, IT security standards, IT security evaluation, and IT architecture).

Dr. Foltz has over a decade of experience working to improve security in information systems. He has degrees in Mathematics, Computer Science, Electrical Engineering, and Strategic Security Studies. He has presented and published research on different aspects of enterprise security, security modelling, and high assurance systems.