

# Enterprise Level Security

Eric D. Trias, *Senior Member IEEE*, William R. Simpson, *Senior Member IEEE*, Kevin E. Foltz, and Frank P. Konieczny<sup>1</sup>

**Abstract**— Mission success and effectiveness depends on timely, secured delivery of information to authorized personnel or systems. Unfortunately, the current security paradigm of building a fortress to protect systems in the network is not sufficient in cyberspace. Further access control measures, such as role-based permissions, to prevent intrusions/disruptions are also insufficient. We present an alternative mission-based approach to a more granular access control paradigm, based on 14 years of research and pilot efforts. This distributed security approach has no need for passwords or system accounts, thus eliminating the associated management overhead. At each step in the authorization process, the system determines validated identities and claims for appropriate access and privileges. The techniques employed are resilient, secure, extensible, and scalable. The system, called Enterprise Level Security (ELS), is currently being researched. This paper discusses the ELS, a web-based security architecture designed to select and incorporate technology into a cohesive set of policies and rules for an enterprise information system. ELS provides application and data level access control automatically, based on the warfighter's current mission profile. As the warfighter's profile changes, authorized accesses are automatically deleted and new ones established to provide relevant, least privileged, mission information at the time of need. The paper begins by introducing ELS, its design principles and architecture, along with its foundational role in developing a forward-looking enterprise baseline; then, it continues by presenting ELS' current status and performance metrics, along with future plans for expansion of capabilities.

**Index Terms**—Access control, authentication, authorization, digital signatures, identity and access management, identity claims, public key infrastructure.

## I. INTRODUCTION

Adversaries continue to penetrate, and in many cases, already exist within our network perimeter, i.e., they have infiltrated the online environment, jeopardizing the confidentiality, integrity, and availability of enterprise information and systems. The fortress model - hard on the outside, soft on the inside - assumes that the boundary can prevent all types of penetration [8]. Proven by a multitude of reported network-related incidents, the previous statements are no longer controversial but a wise assumption for data and information security practitioners. Network attacks are pervasive, and nefarious code is present even in the face of system sweeps to discover and clean readily apparent malware. The focus of this paper is on the security aspects of countering existing known and *unknown* threats based on a

robust identity and access management (IdAM) and how this access control system can dynamically support mission information requirements. There is a working prototype that has been developed and evaluated for security, functionality and scaling issues. Due to space constraints, multi-level security issues are not addressed in this paper.

Enterprise Level Security (ELS) is a capability designed to counter adversarial threats by protecting applications and data with a dynamic attribute-based access control (ABAC) solution. ELS helps provide a high assurance environment in which information can be generated, exchanged, processed, and used. It is important to note that the ELS design is based on a set of high level tenets that are the overarching guidance for every decision made, from protocol selection to product configuration and use [7]. From there, a set of enterprise level requirements are formulated that conforms to the tenets and any high level guidance, policies and requirements.

The basic tenets, used at the outset of the ELS security model are the following:

- |                                    |                                   |
|------------------------------------|-----------------------------------|
| 0. Malicious entities are present. | 8. Need-to-share as               |
| 1. Simplicity.                     | overriding need-to-know.          |
| 2. Extensibility.                  | 9. Separation of function.        |
| 3. Information hiding.             | 10. Reliability.                  |
| 4. Accountability.                 | 11. Trust but verify              |
| 5. Specify Minimal detail.         | (and validate).                   |
| 6. Service-driven rather than a    | 12. Minimum attack surface.       |
| product-driven solution.           | 13. Handle exceptions and errors. |
| 7. Lines of authority should       | 14. Use proven solutions.         |
| be preserved.                      | 15. Do not repeat old mistakes.   |

Current paper-laden access control processes for an enterprise operation is plagued with ineffectiveness and inefficiencies. Given that tens of thousands of government and military personnel transfer locations and duties annually, delays and security vulnerabilities are introduced daily into our operations. ELS mitigates security risks while eliminating much of the system administration required to manually grant and remove user/group permissions to specific applications/systems. Early calculations show that 90-95% of recurring man-hours saved and up to 3 weeks in delay for access request processing will be eliminated by ELS-enabled applications [11]. While perimeter-based architecture assumes that threats are stopped at the front gates, ELS does not accept this precondition and is designed to mitigate many of the primary vulnerability points at the application using a distributed security architecture shown in Figure 1.

## II. ENTERPRISE LEVEL SECURITY

The ELS design addresses five security principles that are derived from the basic tenets:

- Know the Players – this is done by enforcing bi-lateral end-to-end authentication;
- Maintain Confidentiality – this entails end-to-end unbroken encryption (no in-transit decryption/payload inspection);
- Separate Access and Privilege from Identity – this is done by an authorization credential;

<sup>1</sup> This paper submitted for review April 9, 2016. This work was supported by the Secretary of the Air Force, Office of the Chief Information Officer.

Lt Col E. D. Trias, Ph.D, USAF is with SAF/CIO CTO, Pentagon, Washington, DC 20330 USA.

W. R. Simpson, Ph.D, is with the Institute for Defense Analyses, Alexandria, VA 22311 USA.

K. E. Foltz, Ph.D is with the Institute for Defense Analyses, Alexandria, VA 22311 USA.

F. P. Konieczny is the SAF/CIO Chief Technology Officer (CTO), Pentagon, Washington, DC 20330.

- Maintain Integrity – know that you received exactly what was sent;
- Require Explicit Accountability – monitor and log transactions.

Concentrate on the end points.

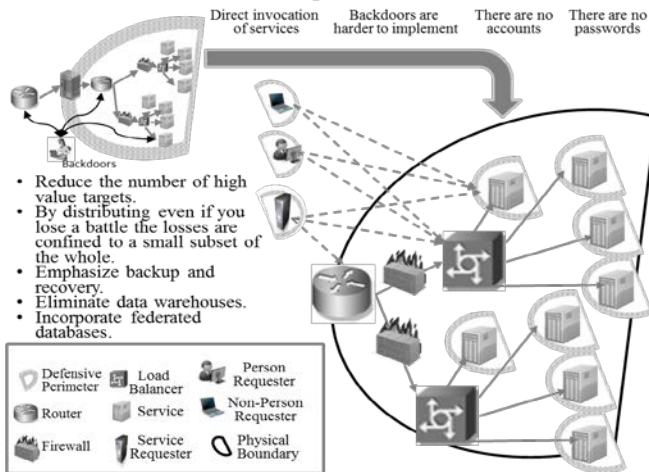


Figure 1: Distributed Security Architecture

#### A. Know the Players

In ELS, the identity certificate is an X.509 Public Key Infrastructure (PKI) certificate [1]. This identity is required for all active entities, both person and non-person, e.g., services, as shown in Figure 2. PKI certificates are verified and validated. Ownership is verified by a holder-of-key check. Supplemental (in combination with PKI) authentication factors may be required from certain entities, such as identity confirming information or biometric data.

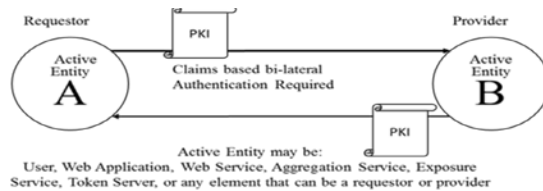


Figure 2: Bi-lateral Authentication

#### B. Maintain Confidentiality

Figure 3 shows that ELS establishes end-to-end Transport Layer Security (TLS) [2] encryption (and never gives away private keys that belong uniquely to the certificate holder).

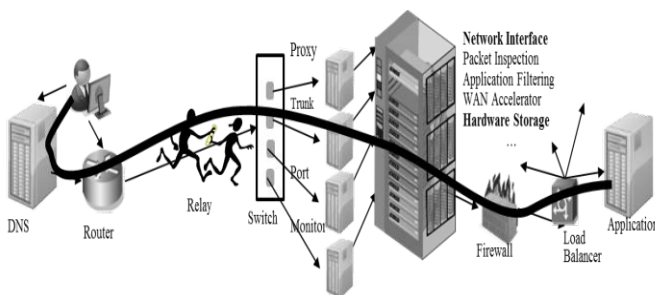


Figure 3: End-to-End Encryption

#### C. Separate Access and Privilege from Identity

ELS can accommodate changes in location, assignment and

other attributes by separating the use of associated attributes from the identity. Whenever changes to attributes occur, claims are recomputed based on new associated attributes (see section III), allowing immediate access to required mission information. As shown in Figure 4, access control credentials utilize the Security Assertion Markup Language (SAML)<sup>2</sup> [3]. SAML tokens are signed, and the signatures are verified and validated before acceptance. The credentials of the signers also are verified and validated. The credential for access and privilege is bound to the requester by ensuring a match of the distinguished name used in both authentication and authorization credentials.

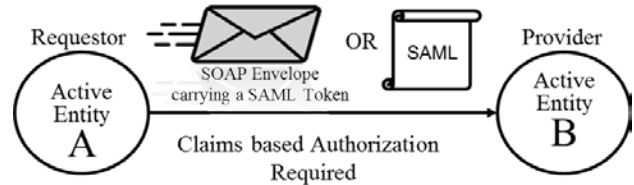


Figure 4: Claims-Based Authorization

#### D. Maintain Integrity

Integrity is implemented at the connection layer by end-to-end TLS message authentication codes (MACs), see Figure 5. Chained integrity, where trust is passed on transitively from one entity to another, is not used since it is not as strong as employing end-to-end integrity. At the application layer, packages (SAML tokens etc.) are signed, and signatures are verified and validated [4].

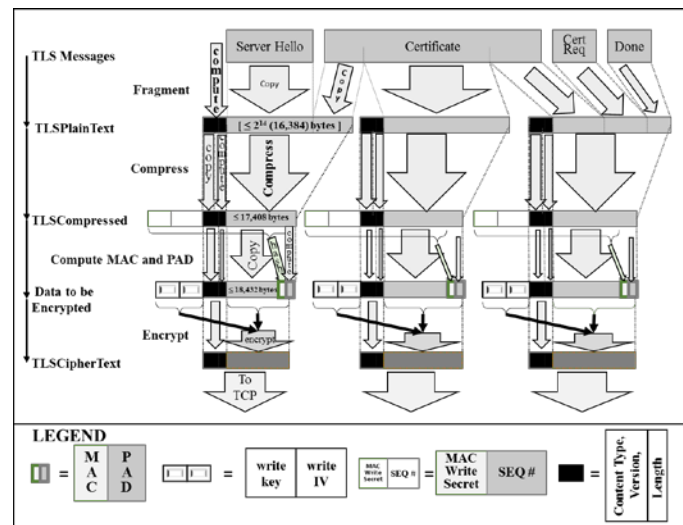


Figure 5: Message Authentication Codes and Other Integrity Measures

#### E. Require Explicit Accountability

As shown in Figure 6, ELS monitors specified activities for accountability and forensics. The monitor files are formatted in a standard way and stored locally. For enterprise files a monitor sweep agent reads, translates, cleans, and submits to an enterprise relational database for recording log records

<sup>2</sup> SAML authorization tokens differ from the more commonly used single-sign-on (SSO) tokens, and in ELS, are not used for authentication.

periodically, or on-demand. Local files are cleaned periodically to reduce overall storage and to provide a centralized repository for help desk, forensics, and other activities. The details of this activity are provided in designated technical profiles and [5, 6].

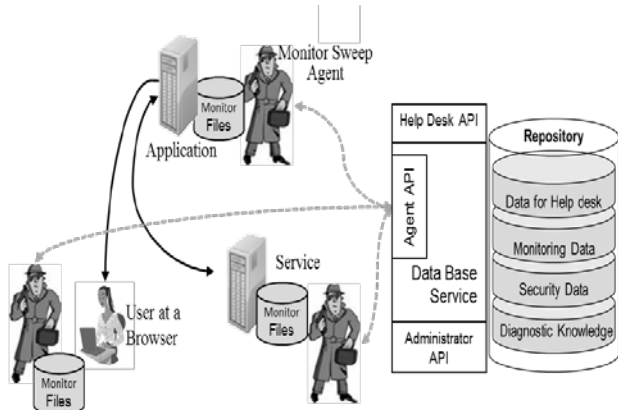


Figure 6: Accountability through Centralized Monitoring

In summary, by abiding with the tenets and principles discussed above, ELS allows users access without accounts by computing targeted claims for enterprise applications (using enterprise attribute stores and asset owner defined claims for access and privilege). ELS has been shown to be a viable, scalable alternative to current access control schemas [11].

### III. ELS ARCHITECTURE

ELS includes functions/capabilities for designated information sharing functions and supporting infrastructure in the enterprise. Many of the key functions that would exist in an enterprise's Target Baseline, see Section IV. Many of these requirements have been documented in technical profiles and use case scenarios. These documents include background information, specific issues with ELS employment, and detailed requirements for use in ELS environments [7].

These services collectively describe the ELS infrastructure that supports many of the security functions across the enterprise in a scalable and automated way. The core of this infrastructure is the Enterprise Attribute Ecosystem. For simplicity, Figure 7 shows only the major components of the system. It consists of a number of services, applications, and data stores that interact in defined ways to provide claims on access to entities within the enterprise based on trusted authoritative data sources for attributes across the enterprise.

The Enterprise Attribute Ecosystem, shown in Figure 7, ingests attributes from various trusted data stores. This creates a central Enterprise Attribute Store that has every attribute populated for each person/non-person entity. This assignment of an attribute from a single authoritative source to an entity reduces human requirements (and potential errors) for maintaining attributes.

The Access Control Registry (ACR) (part of the Service Registry, in lower left corner of Figure 7) contains a set of logical rules for access and privilege to all services and applications that have been registered by their asset owners (Auto Registration Web Service). A claims engine (middle of Figure 7) uses the registered access and privilege rules with the enterprise attributes of an entity to generate the set of

claims that each entity in the enterprise possesses for each of the applications and services. The generated claims are stored in the Claims Repository. This is incrementally updated when attributes change, either periodically or on-demand. The Claims Repository can be distributed for local access and disconnected, intermittent or low bandwidth (DIL) environments. This updating automatically provides the warfighter mission access for his/her mission with minimal delay.

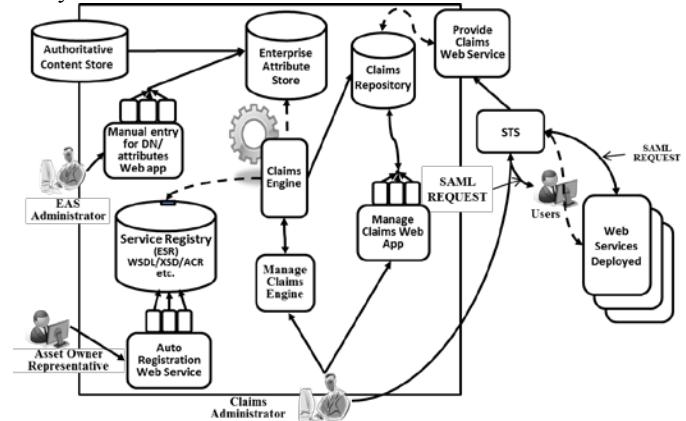


Figure 7: The Enterprise Attribute Ecosystem

A second important infrastructure function in the enterprise is the security token server (STS), see Figure 7. The STS provides SAML tokens to entities in the enterprise, which can be presented for access to services and applications. This is the method by which the claims from the Claims Repository are safely transported to the services and applications that require them.

The STS authenticates the user through mutually authenticated TLS using PKI credentials issued by a trusted issuer. The STS then queries the Provide Claims Web Service for the requesting entity's claims. The STS inserts these claims into a SAML token, digitally signs it, and encrypts the result using the public key of the receiving application.

The application or service then parses the SAML token, validates it using a provided handler code integrated into the web server, and uses the claims to make the access decision. The handler code can be modified only by authorized ELS developers, i.e., not by application/system developers.

Some of the key design decisions in the Enterprise Attribute Ecosystem are the following:

- Separate attribute store and Claims Repository;
- No direct access for external services (those outside of the attribute store ecosystem) to enterprise attribute stores;
- Limit the number of interfaces that support writing to the data stores. No direct access to internal services from outside;
- Provide user convenience functions for claims and attributes as well as asset owner registry services;
- Use end-to-end encryption.

The separation of attributes and claims is important. The attribute store is a sensitive and important data set for the enterprise that must be available and protected. The Claims Repository is a simple mapping of the attribute store through

the ACR. This mapping eliminates many of the Personally Identifiable Information (PII) issues and requirements for handling PII. Another reason to separate attributes and claims is to enable faster access and support DIL operations. For each data center environment, the Claims Repository will be replicated for those applications in that data center; thus does not rely on access to the attributes in the Enterprise Attribute Ecosystem.

Direct access to internal attribute ecosystem services is limited on the attribute input side to the attribute ingestion services (not shown in Figure 7) for each authoritative store. These in turn must be verified and validated before entering information into the Enterprise Attribute Store. On the output side of the attribute ecosystem, the Provide Claims Web Service (upper right of Figure 7) is the only interface, and it simply reads the claims associated with a given user and target application, or service, from the Claims Repository.

Updates to the store are restricted to ensure only valid requests make it to the stores themselves. The attributes are verified and validated before inclusion in the attribute store. Additionally, the ELS system provides other interfaces, such as administrator access and delegation interfaces, but the day-to-day automated interfaces ensure restricted, direct access to the internal stores and services.

A primary security consideration was to reduce the ability of administrators to manually tweak attributes or claims within the EAS. Manual changes are allowed through the *delegation* process which allows short-term creation of manually assigned claims to particular entities. This is meant to address immediate needs, e.g., a subordinate user takes on the role of her supervisor for a short-duration, or until proper attributes can be defined and included in the EAS.

Governance and maintenance of authoritative attribute stores are critical processes that must be in place to ensure a successful ELS enterprise deployment. The inclusion process for attributes is based on a configuration control board that decides whether a proposed attribute maintained in an attribute should be included in the enterprise store and whether its source is sufficiently trusted. To be included in the enterprise store an attribute generally must be used by more than one service or application and in different parts of the enterprise. Or, if needed, and not included, in the Enterprise Attribute Ecosystem, it has to be included through a locally managed store. Exceptions include attributes that are sufficiently security sensitive that a trusted enterprise source is desired or new attributes that are expected to be used more widely in the future. The trustworthiness of the source of attributes is important because all enterprise access decisions are based ultimately on the attributes obtained from these stores.

An important enterprise policy decision is the end-to-end nature of all Transport Layer Security (TLS) connections. Proxies or other intermediaries that impersonate the requester or service provider, even with the endpoint's permission, are not allowed. This includes authentication gateways, proxies, service bus operations, load balancers, and scanners at the server side that decrypt content before passing it to the server itself. Such practices, themselves, introduce vulnerabilities, proxies, and mask the true identities of the communicating parties, i.e., potentially exposing the system to man-in-the-middle/replay attacks.

Mission / Functional Unique Applications and Services	
<b>Computing Services</b> Provide Operating System Services Establish Media Synchronization Functions Provide Audio Production Provide Grid Computing Provide Fault Tolerant Services Provide Cloud Computing Services Provide Virtualization Capabilities Provide Grid Computing Provide Fault Tolerant Services Provide Cloud Computing Services Provide Virtualization Capabilities	<b>Application Foundation</b> Provide Work Flow Services Provide App Hosting Provide Web Hosting Provide Web Services Provide Load Balancing Provide Geographic Info Services
<b>Network Operations/Management</b> Provide Domain Name Services Provide Directory Services Establish Bandwidth Management Perform Traffic Management Provide Spectrum Management Transfer Data via Dynamic Precedence Provide Policy-based Routing Services Execute Network Design Plan Assist Network Design Planner Provide Change Management Services	<b>Data/Information</b> Provide Mediation Services Provide Messaging Services Provide Database Services Provide Data Mining Services Provide Data Mining Services Translate Human Language Compress/Decompress Data Provide Metadata Tagging/Discovery Services
<b>Device Operations/Management</b> Provide Client Device Processing Provide Deployed Personal Edge Devices Provide Automated Info Capture Services Provide Mobile Enterprise Application Platform Services	<b>Network/Communications</b> Provide Network/Precision Timing Provide Streaming Media Multicast Services Establish Programmable Radio Networks Provide Mobile Device Management (MDM) Services Establish Cognitive/Intelligent Spectrum Utilization Manage Info Delivery/Provide Digital Policy w/QoS Provide Application Host Connectivity Services Establish SATCOM Connectivity Establish Mobile Radio Connectivity Provide Tactical Data Link Provide Mobile (Ad Hoc) Networking Provide Telemetry Connectivity Provide Sensor Net Interface Services Control Radio/Terminal Equipment Interface
<b>Enterprise Management</b> Provide Service Desk Capabilities Provide Patch Management Provide Asset Management Provide Asset Discovery Services Provide Problem Management Services Provide Configuration Management Provide Monitoring Services Prioritize System Restoration Perform Trend Analysis Modeling & Simulation	<b>Security</b> Provide Authentication Provide Access Control Provide Basic Security Model Provide Public Key Infrastructure Configure IDPS Provide Cryptographic Services Provide Virtual Private Network Provide Multi-Domain Enclave Security Secure Multi-Level Authentication Protect Data At Rest Protect Data in Processing Provide Network and Application Defense
<b>Application Operations/ Management</b> Provide Application Library/Store Provide Project Management Services Monitor Cloud Virtual Machines Provide Business Process Workflow Execution Services	<b>Storage</b> Provide Consolidate Storage Services
<b>Presentation</b> Provide Presentation Services Provide Widget Services Provide User Defined Operational Picture Provide Web Browsing Provide Automated User Assistance	<b>Enterprise Applications</b> Provide Email Services Provide Collaboration Services Provide Office Automation Services

Figure 8: Target Baseline Reference Model

Requirements for enterprise functions, such as operating systems, databases, messaging, virtualization, are listed in the technical profile for that function. A list of defined enterprise functions is provided in Figure 8. The relevant items for ELS are those contained in the Security block, i.e., Access Control, Authentication, PKI, and Cryptography.

#### IV. ENTERPRISE GOVERNANCE

Deployment of any enterprise solution requires solid documentation and a governance body to ensure proper configuration/standards management. The Technical Profiles (TPs) mentioned above are the core of the documentation process. These define the technologies, standards, and associated requirements for the enterprise. Driving these are Scenario documents that outline the different Use Cases for capabilities that asset owners require from the enterprise. These Scenarios document the functional needs of the asset owners, and the Technical Profile documents, referred to as TPs, define the appropriate technologies, standards, and architectural choices to fulfill those needs. Enterprise users consult the Scenario documents to find a question related to what they want to do, and the Scenario document then refers them to the appropriate TPs to find the answers. The TPs themselves also contain cross-references to each other where appropriate, so a user may have to consult a set of TPs.

The TPs collectively form what is called the Technical Baseline (TB), which contains all technical guidance for the future state of the enterprise. The TB is forward-looking and represents the goal for the enterprise, not the current state. This is necessary to prevent stagnation and to drive the needed change to constantly reassess and improve security. Based on the TB, a set of products is identified to provide functions listed in Figure 8.

The set of products are captured in the Implementation Baseline (IB) documents, and collectively form the IB. The TB drives the IB by providing the enterprise goals, and the IB responds to the TB by providing products, that best implement the functions described in the TB.

ELS continues to be a foundational enterprise capability providing an enterprise-level authorization service. ELS shapes the environment where new and modernizing applications/systems are deployed, and it also determines the level of security protection afforded to each.

## V. CURRENT STATUS

The ELS will reach initial operating capability (IOC) in a production environment in FY16. Major functionalities have been implemented, and initial penetration testing at the National Cyber Range has found no significant architectural problems. Additional detailed vulnerability testing is planned for future test events.

Due to the instantaneous access afforded any authorized user into a system based on the application owner's allowable permissions; authorized users will have immediate access to the application. Within the Air Force alone, it is estimated that system administration requirements would decrease by 90-95% and user delays for access will dwindle to the speed of the updates required by the EAS (from weeks to hours) [11].

Claims generation tests conducted in late 2013 [9] for 1.2 million unique users show that claims may be generated at 215 million generations per hour assuming 119,614 claims being generated with an average time to generate claims of 2.0 seconds and an average claims retrieval time (using the ELS process) of 33ms.<sup>3</sup> These figure are well within the Quality of Service (QoS) expected for this user group.

Scaling tests conducted in mid-2012 [10] indicate that a single STS can handle 800 SAML tokens/sec, which is 50,000/min, or 250,000 every 5 minutes. A rate of one STS request every 5 minutes per user is the maximum anticipated (peak sustained rate), then 4 STSs per 1,000,000 users in the enterprise, which is very reasonable. For redundancy, locality, and surges, and load balances losses a planning figure of 10 STSs per 1,000,000 entities is used. This is very achievable and can easily scale to larger enterprises. The application handler code to process SAML tokens has been generated for inclusion with .Net and Java applications and services. It has also undergone initial testing.

These test results were documented as the result of a carefully crafted spiral development process, see Figure 9. The figure outlines ELS milestones beginning with initial research, in 2002, along with the drafting of the design tenets and development of each of the major components. These include:

- Fully encrypted unbroken end-to-end communications (TLS with message authentication codes).
- Bi-lateral PKI authentication for all enterprise entities;

<sup>3</sup> Test Setup includes load test Servers 2 (12 Cores total); total concurrent user load 1000 (no waits); claim set contains 3 claims; 2 mapped attributes; 1 computed claims. Server configuration includes all virtualized servers; data server: 8 core / 32GB RAM; claims server: 8 core / 32 GB RAM.

- SAML-based approaches for access and privilege. The SAML creation and utilization are hardened for vulnerability mitigation;
- Embedded SAML handles for consistency in application;
- Claims-based access and privilege approach as opposed to attributes and roles;
- Defined federation and delegation processes;
- Virtualization inspection handlers (in process).

A full implementation began in 2012 with a spiral based roll-out leading to pathfinder applications, testing and evaluations, and JIE application currently in process.

## VI. WAY AHEAD

ELS will continue to evolve as the TB matures and iterates through Scenario and Technical Profile documents. ELS is one of its foundational service as implementation throughout the Air Force proceeds to capture enterprise use cases and define their associated technical solutions. As baselines are established, evaluation of various applications from the other components and environments, such as C2 and tactical, will be accommodated to fine tune ELS to meet warfighter needs.

The Enterprise Attribute Ecosystem development will continue, and with testing and other feedback it will be hardened and operationalized for enterprise operation. Other elements of the ELS, including the handler code installed on the servers will be hardened, according to DoD policies, and provided to developers of new applications and services. Application and service developers will be integrated into the process so that they understand what is expected with ELS, and assistance will be provided through hands-on support and increased documentation of the ELS process.

## VII. CONCLUSION

With the current high operations tempo, warfighters must have the information and access to the systems they require to execute their missions at the time of need. Current authorization paradigm requires a cadre of highly privileged administrators to maintain user account permissions for every system and data sources required. Human errors, delays in request processing, and credential misuse add to the enormous risks our warfighters are facing in their day-to-day activities. Further aggravating the challenges to successful mission execution, and future operations, is the determined presence of malicious actors in the contested environment [8]. We must continue to advance our security posture by protecting the applications and data at the source. It is in this vein that ELS was conceived, i.e., to provide a superior way for a secure, scalable *access control* for the enterprise.

The ELS architecture, based on core security tenets, reflect the overall goals and philosophy of the enterprise and its security. From these tenets, requirements are derived for core security operations and other functional goals to support information sharing within and outside the enterprise.

The techniques employed are resilient, secure, extensible, and scalable. ELS has been tested and is mature in its development at this time and ready to become the security architecture to become a major component of its IdAM solution.



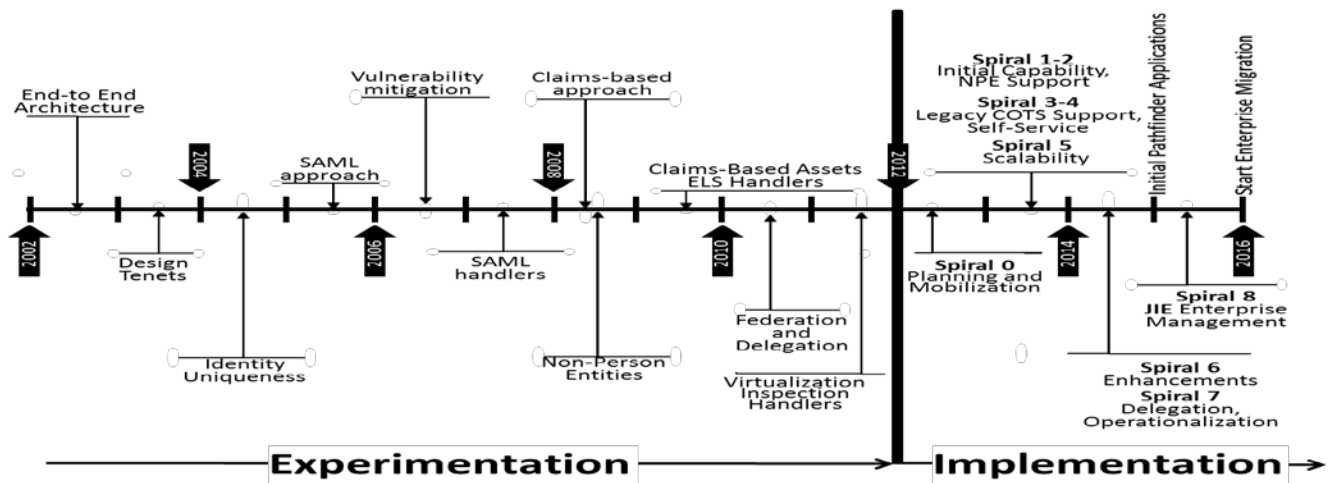


Figure 9: ELS Evolution

This paper discussed ELS, a web-based security architecture designed to select and incorporate technology into a cohesive set of policies and rules for an enterprise information system. ELS provides application and data level security and is presented as viable, scalable alternative to current access control management saving thousands of system administration man-hours. Initial calculation [11] shows that the initial standup of ELS will cost approximately 75% of the annual recurring costs as compared to the current process using DD Form-2875 [12].

The documentation of the Target Baseline will bring together the current detailed operation of ELS-compliant systems with the products in use and the enterprise functions that they are implementing [7]. This methodology provides a resilient approach to providing and maintaining security for a large enterprise in an automated, scalable, and modifiable way.

#### REFERENCES

- [1]. X.509 Standards
  - a. DoDI 8520.2, Public Key Infrastructure (PKI) and Public Key (PK) Enabling, 24 May 2011
  - b. JTF-GNO CTO 06-02, Tasks for Phase I of PKI Implementation, 17 January 2006
  - c. X.509 Certificate Policy for the United States Department of Defense, Version 9.0, 9 February 2005
  - d. FPKI-Prof Federal PKI X.509 Certificate and CRL Extensions Profile, Version 6, 12 October 2005
  - e. RFC Internet X.509 Public Key Infrastructure: Certification Path Building, 2005
  - f. Public Key Cryptography Standard, PKCS #1 v2.2: RSA Cryptography Standard, RSA Laboratories, Oct 27, 2012
  - g. 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
- [2]. TLS family Internet Engineering Task Force (IETF) Standards
  - a. RFC 2830 Lightweight Directory Access Protocol (v3): Extension for Transport Layer Security, 2000-05
  - b. RFC 3749 Transport Layer Security Protocol Compression Methods, 2004-05
  - c. RFC 4279 Pre-Shared Key Ciphersuites for Transport Layer Security (TLS), 2005-12
  - d. RFC 5246 The Transport Layer Security (TLS) Protocol Version 1.2, 2008-08
  - e. RFC 5289 TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode (GCM), 2008-08
  - f. RFC 5929 Channel Bindings for TLS, 2010-07
  - g. RFC6358 Additional Master Secret Inputs TLS, 2012-01
  - h. RFC 7251 AES-CCM Elliptic Curve Cryptography (ECC) Cipher Suites for TLS, 2014-06
  - i. RFC 7301 Transport Layer Security (TLS) Application-Layer Protocol Negotiation Extension, 2014-07
  - j. RFC 7457 Summarizing Known Attacks on Transport Layer Security (TLS) and Datagram TLS (DTLS), 2015-02
- [3]. Organization for the Advancement of Structured Information Standards (OASIS) open set of Standards
  - a. N. Ragouzis et al., Security Assertion Markup Language (SAML) V2.0 Technical Overview, OASIS Committee Draft, March 2008
  - b. P. Mishra et al. Conformance Requirements for the OASIS Security Assertion Markup Language (SAML) V2.0. OASIS Standard, March 2005.
  - c. S. Cantor et al. Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.0, OASIS Standard, March 2005
- [4]. William List 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.
- [5]. William R. Simpson and Coimbatore Chandrasekaran, CCCT2010, Volume II, pp. 84–89, "An Agent Based Monitoring System for Web Services," Orlando, FL, Apr 2011.
- [6]. William R. Simpson and Coimbatore Chandrasekaran, 1st International Conference on Design, User Experience, and Usability, part of the 14th International Conference on Human-Computer Interaction (HCII 2011), "A Multi-Tiered Approach to Enterprise Support Services," 10 pp. Orlando, FL, July 2011. Also published in: A. Marcus (Ed.): Design, User Experience, and Usability, Pt I, HCII 2011, LNCS 6769, pp. 388–397, 2011. © Springer-Verlag Berlin Heidelberg 2011.
- [7]. Technical Profiles for the Consolidated Enterprise IT Baseline, release 3.0 available at (CAC required) (currently working 4.0): <https://intelshare.intelink.gov/sites/afceit/TB>
- [8]. Frank Konieczny, Eric Trias and Nevin Taylor, "SEADE: Countering the Futility of Network Security," Air and Space Power Journal, Sep-Oct 2015, Vol 29, No.5, pg. 4.
- [9]. Briefing prepared by Accenture Corporation, "USAF Enterprise Level Security, Spiral 5, Codeless Migration of Legacy .NET Applications, High Performance Claims Engine and Performance Test Results", dated 27 September 2013.
- [10]. Email from Michael Leonard, MITRE Organization on behalf of USAF AFMC ESC/HNCDDD, dated May 10, 2012, Subject: "Performance / Scalability"
- [11]. Email from Rudy Rihani, Project Manager, Accenture Corporation, dated March 6, 2016, Subject: "manpower savings with ELS"
- [12]. DD-2875, SYSTEM AUTHORIZATION ACCESS REQUEST (SAAR), Executive Order 10450, 9397; Public Law 99-474.