Insider Threat Metrics in Enterprise Level Security

William R Simpson, *Member IAENG*, and Kevin E. Foltz

***Abstract*— Enterprise Level security (ELS) is an application security model that has no accounts or passwords, and consequently identity is an important issue. All person and non-person entities in ELS are registered and known. PKI credentials are issued, and when necessary, multi-factor authentication is used to improve the assurance of the identity. Because the next step in ELS is claims-based access and privilege, many data owners are worried about the trustworthiness (sometimes called reputation) of the identified requesters (this applies to person and non-person entities within the enterprise). Individuals are vetted periodically, and a baseline is established by those instances; however, activities that occur between those vetting events may provide clues about the trustworthiness of the individuals. Similarly, pedigrees in software and hardware entities are established periodically. Because the terms trust and integrity are overloaded, we refer to these data as veracity. Further, when requested, the veracity that applies to certain categories will be provided as counter-claims along with the claims. These counter-claims may be used by the applications and services for increased levels of surveillance and logging and perhaps even limitation of privilege. The computation of veracity brings about security concerns and requires special handling. This paper reviews the data categories, data requirements, security issues, and data resources that apply to entity veracity, as well as the counter-claim structures and issues associated with their tracking and usage. The paper then presents findings and recommendations, along with the future work necessary to complete this evolution.**

***Index Terms* — Behavior, Claims, Counter-Claims, Insider Threat, Integrity, Reputation, Motivation, Veracity**

I. INTRODUCTION

Guidance and policies for insider threat are incomplete as of this time. Insiders may be either loyal but careless members of the enterprise, or malicious (nefarious) in their intent. Training and some limited mechanisms such as white or black listing are used with the former. Monitoring of activities is primarily for forensics. The nefarious insider may know all of the monitoring and avoid their intent. A second area of veracity (sometimes called reputation) can provide a measure of susceptibility to co-option or conversion to nefarious activities. This paper is based in part on a paper published by WCECS 2017 [1].

Manuscript received 27 August 2018; revised 10 September 2018.

This work was supported in part by the U.S. Secretary of the Air Force and The Institute for Defense Analyses (IDA). The publication of this paper does not indicate endorsement by any organization in the Department of Defense or IDA, nor should the contents be construed as reflecting the official position of these organizations.

Kevin E. Foltz is with the Institute for Defense Analyses. (email: kfoltz@ida.org)

William R. Simpson is with the Institute for Defense Analyses, 4850 Mark Center Drive, Alexandria, Virginia 22311 USA and is the corresponding author phone: 703-845-6637, FAX: 703-845-6848 (e-mail: rsimpson@ida.org)

The insider threat is required to be monitored and assessed, especially for those government enterprises subject to presidential executive orders. Since a number of malicious insiders such as Edward Snowden [2], Bradley Manning [3], and others [4], we have no choice but to assess our own insider threat situation. An insider threat is:

> "… a malicious threat to an organization that comes from people within the organization, such as employees, former employees, contractors or business associates, who have inside information concerning the organization's security practices, data and computer systems." [5]

The manifestation of the threat may come from any entity in the environment, person or non-person. The spate of insider activity has led to a U.S. executive order [6] that requires, in part, federal agencies and enterprises to:

> "…perform self-assessments of compliance with policies and standards issued pursuant to sections 3.3, 5.2, and 6.3 of this order, as well as other applicable policies and standards, the results of which shall be reported annually to the Senior Information Sharing and Safeguarding Steering Committee established in section 3 of this order…."

For Enterprise Level Security (ELS) [7] federal applications, we must include these self-assessments. The requirement has led to the development of new products and an overwhelming volume of white papers and other research telling us how some vendors would do this assessment, and a number of patents pending [8-11]. All of this leads to a number of product offerings to perform the analysis of entity veracity within the enterprise. A summary of these techniques (through 2011) is provided in [12]. The basic idea is to gather information concerning the trustworthiness of an entity in our system.

II. INTEGRITY, REPUTATION, & VERACITY

Generally, the determination of trustworthiness of an individual is based upon an assessment of the integrity of that individual. One definition of integrity is given below:

> "Integrity is the quality of being honest and having strong moral principles; moral uprightness. It is generally a personal choice to hold oneself to consistent moral and ethical standards. In ethics, integrity is regarded by many people as the honesty and truthfulness or accuracy of one's actions." [13]

Social media would define this as reputation, which is good because integrity is already over-used in the information technology (IT) literature. However, the literature defines reputation as a soft issue.

> "Reputation is the estimation in which a person or thing is held, especially by the community or the public generally." [14]

Microsoft has refined reputation by adding trust:

> "Reputation Trust represents a party's expectation that another party will behave as assumed, based upon past experience. Reputation Trust is bidirectional and can be split into Consumer Reputation Trust and Provider Reputation Trust." [15]

But trust is an overloaded term in information technology and requires a great deal of context. The dictionary description of veracity comes closer to the target, and it is not used in any of the IT contexts associated with ELS:

> "Veracity is the quality of being truthful or honest." [16]

From the IT standpoint, we have adopted the concept of veracity and tailored its definition to be more amenable to self-assessment in ELS environments:

Entity Veracity is the degree to which an entity is worthy of trust as demonstrated by resistance to or avoidance of factors that denigrate trust or compromise reliability. Positive factors may enhance veracity, and negative ones may reduce veracity. Veracity is based upon recognized accomplishments and failures, along with the associated stress factors or other trust debilitating factors present. A history of actions in difficult circumstances provides strong evidence for or against veracity.

The next step is to determine which of the factors need to be measured. But first we need to understand how identity and access control are handled within ELS.

III. ENTERPRISE LEVEL SECURITY

Security Process Background

This work is part of a body of work for high-assurance enterprise computing using web services. The process has been developed over the last fifteen years and is termed ELS.

In certain enterprises, the network is continually under attack. Examples might be:

- Banking industry enterprise.
- Defense industry applications,
- Credit card consolidation processes.
- Commercial point-of-sale processes.
- Medical -- privacy and statutory requirements,
- Content Distributor's -- rights in data, theft of content.

The attacks have been pervasive and often include previously unseen attack vectors and they continue to the point that nefarious code may be present, even when regular monitoring and system sweeps clean up readily apparent malware. This omnipresent threat leads to a healthy paranoia of many threats including resistance to observation, intercept and masquerading. The web interface is the best way to provide access to many of its users despite this highly active threat environment.

One way to maintain capability in this type of environment is to not only know and vet your users, but also your software and devices. Even that has limitations when dealing with the voluminous threat environment. Today we regularly construct seamless encrypted communications between machines through SSL or other TLS. These do not cover the "last mile" between the machine and the user (or service) on one end, and the machine and the service on the other end. This last mile is particularly important when we assume that malware may exist on either machine, opening the transactions to exploits for eaves dropping, ex-filtration, session high-jacking, data corruption, man-in-the-middle, repeat replay, masquerade, blocking or termination of service, and other nefarious behavior.

ELS is a capability designed to counter adversarial threats by protecting applications and data with a dynamic claims-based access control (CBAC) 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 [17]. From there, a set of enterprise level requirements are formulated that conforms to the tenets and any high level guidance, policies and requirements.

ELS has evolved over the last fifteen years. Many experiments have led to a few successes. They roughly followed the path shown below (but not without backtracking and reconfiguring). The material below is for context and the evolution is broken into two general areas of experimentation and implementation. Many possible implementations exist and the current implementation may be modified for many reasons, including the dynamic nature of the cybersecurity environment.

Figure 1 provides some details of the evolution during the experimental years. Each of the icons has had one or more software and configuration experiments in their development. Figure 2 provides the implementation through the subject of this paper. Many of these have had experiments and some are awaiting instantiation, such as the homomorphic computing. The implementation has gone through a spiral development and has been documented extensively for procurement, including requirements, by the fifth evolution of the Consolidated Enterprise IT Baseline (CEITB) [18] consisting of 63 documents (38 technical profiles and 25 scenario documents).

Almost all of the topics and milestones in the two figures are published in the open literature. These are web- or google scholar- searchable, and provide a large number of details for the interested reader. It is not the purpose of this paper to provide the complete detail and history of ELS, but to provide the context that has led to the evaluation of entity veracities. This evolution continues today.

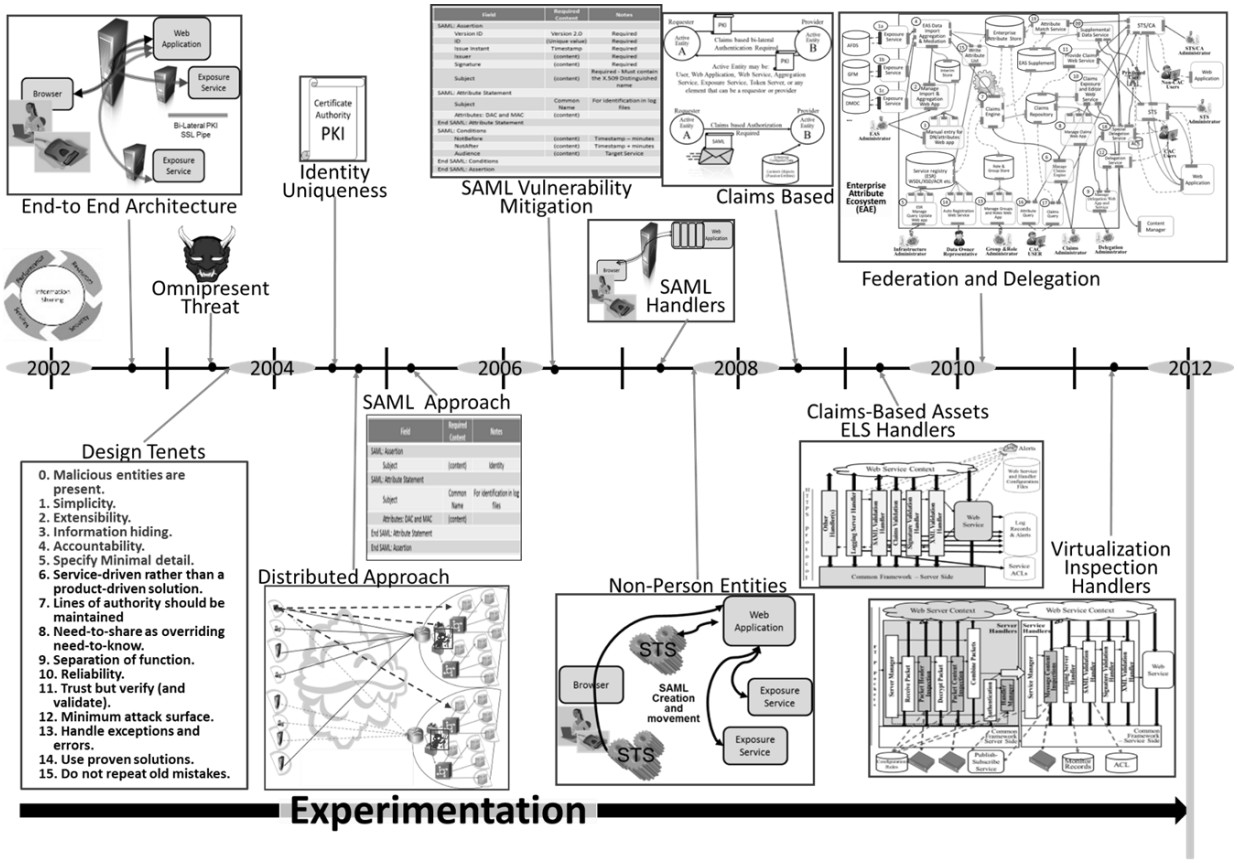

Fig 1 ELS Experimental Evolution

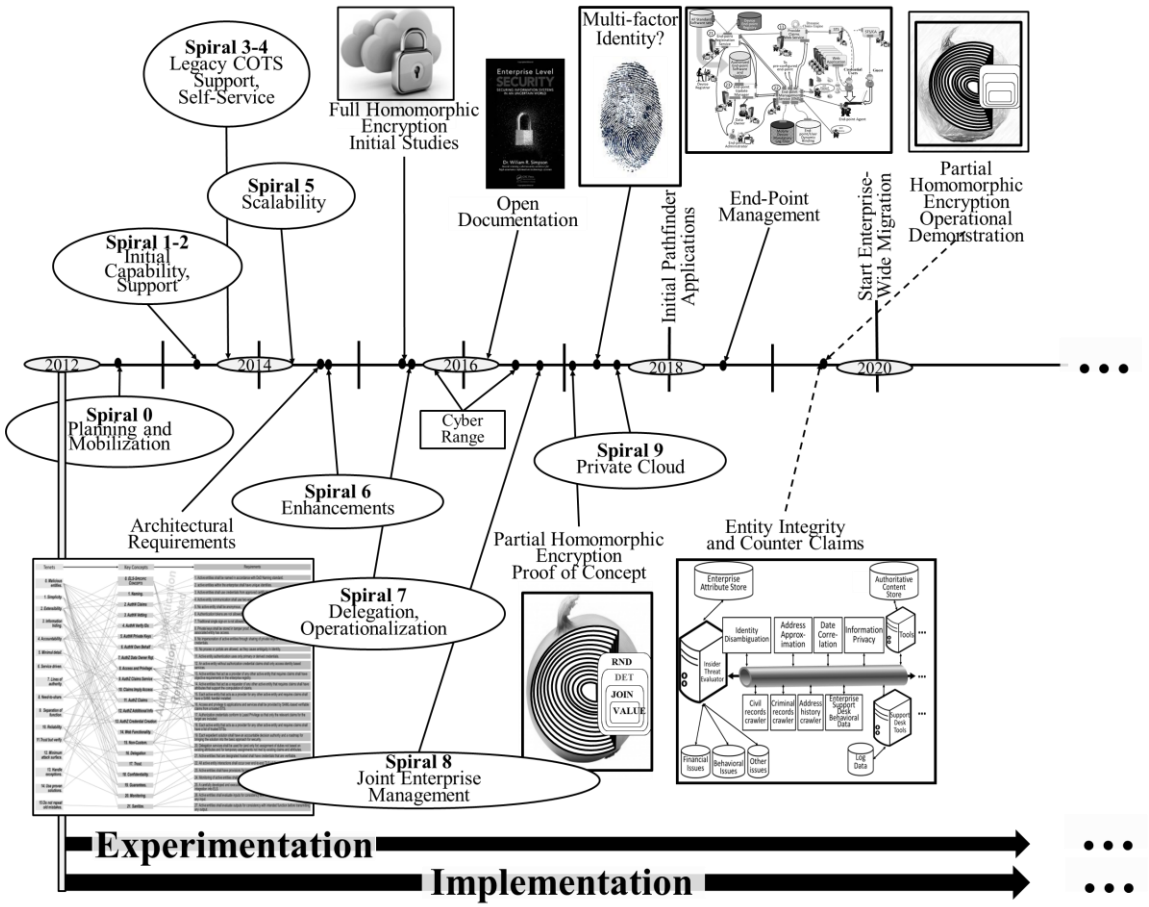

Fig 2 ELS Implementation Evolution

Design Principles

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

0. The ***zeroth*** tenet is that the *malicious entities are present* and can look at network traffic and may attempt to modify that traffic by sending virus software to network assets. Current threat evaluation indicates that attacks are often successful at all levels; discovering these attacks and their consequences is problematic. In many cases attackers may compromise and infiltrate before a vulnerability can be mitigated by software changes (patches).

1. The ***first*** tenet is *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. The ***second*** tenet, and closely related to the first, is *extensibility*. Any construct we put in place for an enclave should be extensible to the domain and the enterprise, and ultimately to cross-enterprise and coalition.

3. The ***third*** tenet is *information hiding*. Essentially, information hiding involves only revealing the minimum set of information to the requester and the outside world needed for making effective, authorized use of a capability.

4. The ***fourth*** tenet is *accountability*. In this context, accountability means being able to unambiguously identify and track what active entity in the enterprise performed any particular operation (e.g., accessed a file or IP address, invoked a service). Active entities include people, machines, and software process, all of which are named registered and credentialed. By accountability we mean attribution with supporting evidence.

5. This ***fifth*** tenet is *minimal detail* (to only add detail to the solution to the required level). This combines the principles of simplicity and information hiding, and preserves flexibility of implementation at lower levels.

6. The ***sixth*** is the emphasis on a *service driven* rather than a product-driven solution whenever possible. Services should be separated as stated in the separation of function tenant. This also allows simplification and information hiding.

7. The ***seventh*** tenet is that *lines of authority* should be preserved and information assurance decisions should be made by policy and/or agreement at the appropriate level. An example here is that data owners should implement sharing requirements even when the requirements come from "higher authority."

8. The ***eighth*** tenet is *need-to-share* as overriding the need-to-know. Often effective health, defense, and finance rely upon and are ineffective without shared information. Shared does not mean released and the differences must be clear. However, judicious use of release authority and delegated access lead to a broader distribution of information. This leads to a more formalized delegation policy both within and outside of the enterprise.

9. The ***ninth*** tenet is *separation of function*. This makes for fewer interfaces, easier updates, maintenance of least privilege, reduced and easier identified vulnerabilities and aids in forensics.

10.The ***tenth*** tenet is *reliability*; security needs to work even if adversaries know how the process works. In setting up a large scale enterprise we need to publish exactly how things work. Personnel, computer operations people and vendors need to know how the system works and this should not create additional vulnerabilities.

11.The ***eleventh*** tenet is to *trust but verify* (and validate). Trust should be given out sparingly and even then trusted outputs need checking. Verification includes checking signature blocks, checking that the credential identities match (binding), checking the time stamps, checking to whom information is sent. Checking information received is identical to information sent, etc. Validation includes checking issuing authority, checking certificate validity, checking identity white lists and black lists.

12.The ***twelfth*** tenet is *minimum attack surface*; the fewer the interfaces and the less the functionality in the interfaces, the smaller the exposure to threats.

13.The ***thirteenth*** tenet is *handle exceptions* and errors. Exception handling involves three basic aspects. The first is logging. The second is alerting and all security related events should be alerted to the Enterprise Support Desk (ESD). The third is notification to the user.

14.The ***fourteenth*** tenet is to *use proven solutions*. A carefully developed program of pilots and proofs of concepts has been pursued before elements were integrated into ELS. It is our intention to follow that process even when expediency dictates a quicker solution. Immediate implementation should always be accompanied by a roadmap for integration that includes this tenet.

15.The ***fifteenth*** tenet is *do not repeat old mistakes*. From a software point of view, this has many implications. First, never field a software solution with known vulnerabilities and exploits. There are several organizations that track the known vulnerabilities and exploits and an analysis against those indexes should be required of all software. Second, a flaw remediation system is required. After a vulnerability analysis, fixes may be required, after fielding, fixes will be required as new vulnerabilities and exploits are discovered. Third, from an operations standpoint take time to patch and repair, including outputs from the flaw remediation and improvements in Security Technical Implementation Guidelines.

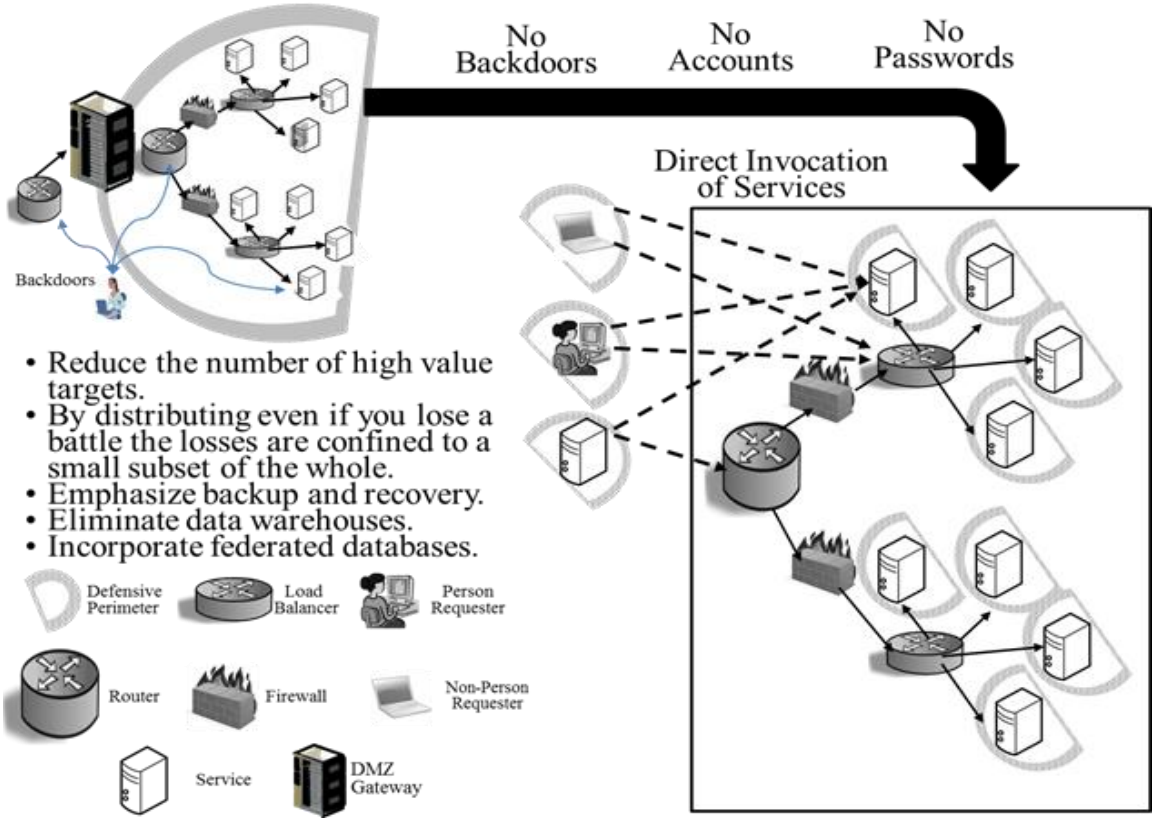

Fig 3 Distributed Security Model

Current paper-laden access control processes for an enterprise operation are plagued with ineffectiveness and inefficiencies. Given that in a number of enterprises tens of thousands of personnel transfer locations and duties annually, delays and security vulnerabilities are introduced daily into their operations.

ELS is an application security model that is able to mitigate security risk while eliminating much of the system administration required to manually grant and remove user/group permissions to specific applications/systems. Early calculations show that for government and defense 90-95% of recurring man-hours are saved and up to 3 weeks in delay for access request processing are eliminated by ELS-enabled applications [19]. While perimeter-based models assume 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 model shown in Figure 3.

Security Principles

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

Know the Players

The ELS application security model requires that every application level entity in the environment have a unique identity that may be verified and validated. In ELS, the identity certificate is an X.509 Public Key Infrastructure (PKI) certificate [20]. This identity is required for all active entities, machines, persons and non-persons, e.g., devices, services, as shown in Figure 4. PKI certificates are verified and validated. Ownership is verified by a holder-of-key (HoK) check.

The authentication is bi-lateral so that each participant in a transaction is aware of the other participant. Supplemental (in combination with PKI) authentication factors may be required from certain entities, such as identity confirming information or biometric data. Specifically mobile devices and those requiring higher levels of assurance. A specific exemption to this is that temporary certificates for enterprise personnel who do not have an available PKI token for a certificate may be obtained using multi-level authentication. Because of the vulnerabilities associated with software PKI certificates, the life is limited. This also makes revocation checking ineffective, so no revocation checking is performed. This certificate has a short life, but allows an individual to set up sessions with one or two applications before it expires.

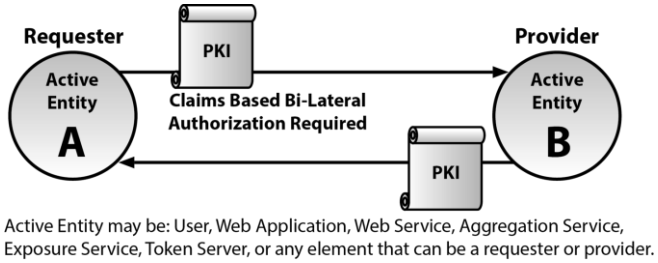

Fig 4 Bi-lateral Authentication

Maintain Confidentiality

Figure 5 shows that ELS establishes end-to-end Transport Layer Security (TLS) [21] encryption. The private keys that belong uniquely to the certificate holder are held in hardware storage: Personal Identity Verification (PIV) type cards with embedded chips, for individuals, and Hardware Storage Modules (HSM)s for hardware and software entities.

The private keys are only accessed by the holder and the keys are never shared with network appliances or other entities. The encryption must remain unbroken through service hardware such as routers, firewalls, and load balancers. There are no delegates or proxies that can be used as masquerades.

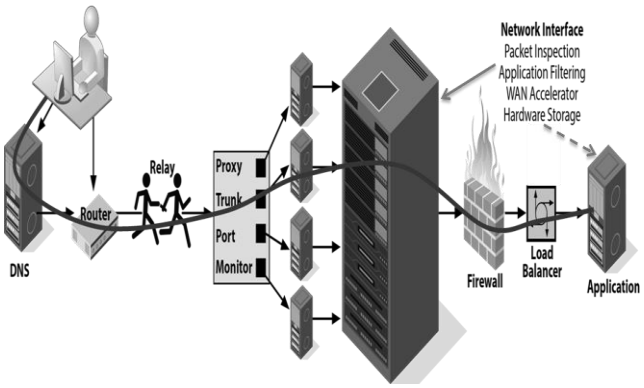

Fig 5 End-to-End Encryption

Separate Access and Privilege from Identity

The separation of identity and access and privilege allows for the breaking of the account paradigm that is the subject of many vulnerabilities. It also allows for the automation of provisioning employees on the move with access and privilege commensurate with their new assignments. 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, allowing immediate access to required applications.

As shown in Figure 6, access control credentials utilize the Security Assertion Markup Language (SAML). SAML was chosen because it has many equivalent properties to the PKI identity certificate. The tokens may be verified (by signature checking) and validated (by checking for trusted issuers). SAML authorization tokens differ from the more commonly used single-sign-on (SSO) tokens, and in ELS, SAML tokens are not used for authentication. [22].

SAML tokens are created and signed by a Security Token Server (STS). 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 identity used in both authentication and authorization credentials.

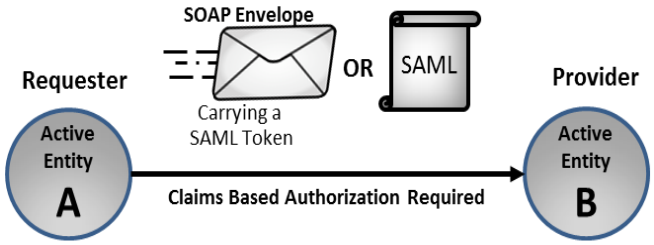

Fig 6 Claims-Based Authorization

Maintain Integrity

In all cases, integrity in communication means that the message that was received was identical to the message sent (no additions, deletions or modifications). Integrity is implemented at the connection layer by end-to-end TLS message authentication codes (MACs), see Figure 7. 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 [23].

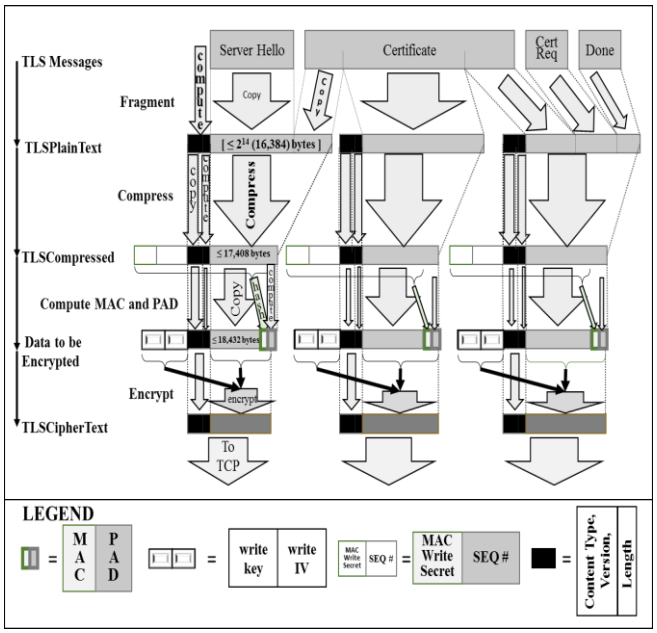

Fig 7 Integrity Measures

Require Explicit Accountability

All active entities within ELS are required to act on their own behalf (no proxies or impersonation allowed). As shown in Figure 8, 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 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 [24, 25].

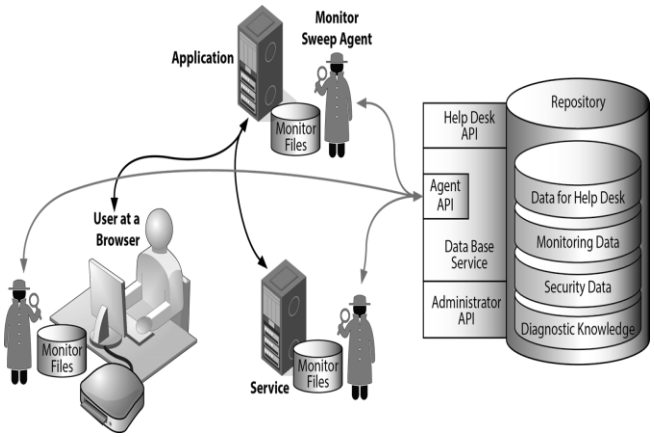

Fig 8 Accountability Monitoring

IV. MEASURING VERACITY

Figure 9 shows the security issues associated with just the computation of an entity's trustworthiness. Access to public records and sources that are not vetted opens vulnerabilities not tolerated in a high assurance environment such as ELS. The initial implementation will be done in isolation from the enterprise and data will be ported to the enterprise. The figure shows the desired ultimate architecture where the computation is isolated and the enterprise may be provided a read only interface of the results (which may initially be a mirror of the actual veracity store). Two additional concerns in the figure include a read only interface from the computation environment to the enterprise attribute store (which may initially be a mirror of the actual enterprise attribute store), and a read only interface from the computation environment to the enterprise support desk behavioral data (which may initially be a mirror of the actual enterprise support desk behavioral data). Paranoia is warranted when dealing with unclean data and the entire insider threat analysis system will be heavily monitored, and sanitized often with complete software re-installation at periodic intervals. Several steps should be taken to isolate the veracity computation from the enterprise.

• Openness. Let enterprise stakeholders know the process and effects that they will encounter for the protection of their data as well as your data and resources.

• Policy. Establish enterprise policy on veracity usage in order to shape each of the bullets below and provide for the bullet above .

• Isolation. Keeping the veracity system isolated from enterprise resources that may be corrupted or abused is paramount. This can be done by setting up a de-militarized Zone (DMZ). The DMZ is disconnected from the enterprise except during times of refreshing. In the DMZ will reside mirrors of enterprise data and services that are needed, but these are not linked back to the enterprise. These mirrors are periodically (like overnight or more or less frequently depending on the business model) refreshed from enterprise resources. Less frequently, the services themselves are rebuilt from enterprise resources [28]. Figure 9 shows the initial setup of the Veracity System.

• Transactional. The veracity system interactions are recorded on a transactional basis and then executed against the DMZ data bases. The online data may have a delay of (notionally 24 hours) to reflect these transactions[29].

• Analyze. Record and analyze the usage and sources.

• Cleaning and Reviewing. During the refresh, the transactions are cleansed and reviewed for nefarious behavior. Those that pass muster are imported into the database that is accessed by the enterprise and executed against the enterprise data bases. Those that do not result in an alert to the security personnel that the transaction was rejected [30].

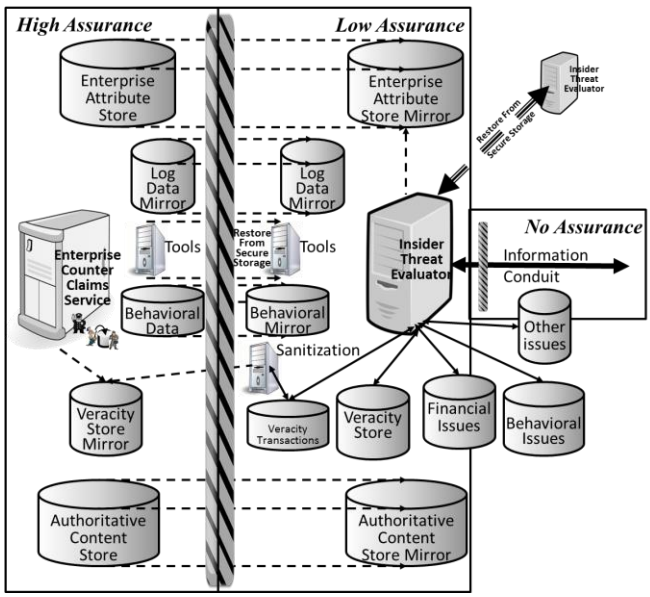

Fig 9. Insider Threat Setup.

The figure shows the data mirroring required:

- Relavent information from the enterprise attribute store, used for data correlation for each of the unique identities in the enterprise.

- Relavent log data from enterprise activities, as well as log data from veracity computations for both analysis and later forensics when required.
- Relavent behavioral data from activities within the enterprise to be included in the veracity analyses.
- The veracity store itself for use in the enterprise.

Data mirroring will occur on an exception basis and over a short period such as every twenty-four hous, but it should be configurable across a wide range of activity.

Additionally, the analysis tools (including the Insider Threat Evaluator) need to be replicated in the veracity system because it will be disconnected from the enterprise when computing veracity data. These tools include:

- Identity disambiguation – resolving names and other identity metrics when matches are less than perfect.
- Data Correlation between current and old addresses and other confirming data.
- Information privacy elements (such as social security numbers, etc.) may be used for correlation, but need to marked for special access only and not included in normal reporting.

Software activity is monitored, suspicious activities will lead to forensics, software will be modified to include mitigations and re-generated. However, new and unfamiliar intrusions and nefarious invasions may take some time to sort out.

All software in the veracity system is periodically refreshed from secure memory and reconfigured to avoid as-yet undetected threat activity. The periodicity of this refresh is configurable, and may be more frequent during high threat activity. The second fence (at the information conduit) will be activated during refresh and the system will be disconected from the information conduit. The figure also shows clearly the denotation of assurance assigned to each zone. Data from the low assurance zone never leaves that zone with the exception of the sanitized transactional data that is used to update the enterprise veractiy store mirror.

Figure 10 provides a brief depiction of the information conduit flow. In the figure the information is imported from a variety of configured sources along the bottom of the conduit. These sources may change from time to time as more reliable or accurate sources are identified.

During normal operation, the system is disconnected from the enterprise and any changes to the veracity database are recorded as transactions. These transactions will be sanitized before updates are accepted on the next refresh cycle. Because the reliability and accuracy of data may be less than satisfactory in many instances, the top half of the information conduit is devoted to tools for correlation of data. In many open sources correlation by name, residence and other vital statistics are less than reliable and veracity metrics must be at least reasonably verified. In many cases, the actions will be discussed and issues clarified or removed when mistakes are made. Information not deemed unreliable, but not totally verified, will be discounted in its effect. This issue is further described under appropriate categories.

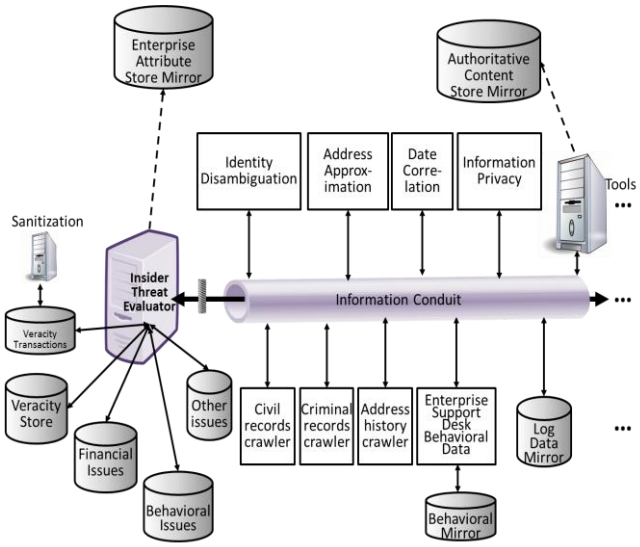

Fig10. Insider Threat Information Conduit

Figure 11 shows the notional system in its up and running configuration without the details of mirror and sanitization operations. The flows are only partially complete in that each pieces of information obtained from the information sources on the lower side of the conduit, must be subjected to the correlation activities on the upper half of the information conduit. Further, any information that makes a change in the veracity store will be recorded as a transaction for later sanitization (as shown in Figure 9). Only after sanitization will the enterprise veracity store be updated.

We have presented a form of self-assessment that evaluates veracity from the ELS application security model perspective rather than from the perspective of the product's baseline. This paper also addresses the issues associated with the self-assessment, and it provides a framework and a process for using veracity information within the ELS application security model. To do this, we examine integrity, reputation, and veracity as they apply to the problem of the insider threat. A list of indicative events may be formulated by category and data sources. [26–27]

We start categorization with person entities because this is required in the self-assessment, but veracity extends to all entities within the enterprise because non-person entities may actually be under insider threat control. For all entities, we assume a default value of 1.0 for veracity before detailed veracity computations are made. This is the minimum value needed to pass periodic re-evaluations, so it is assumed that all entities in the enterprise possess this value unless veracity factors indicate otherwise.

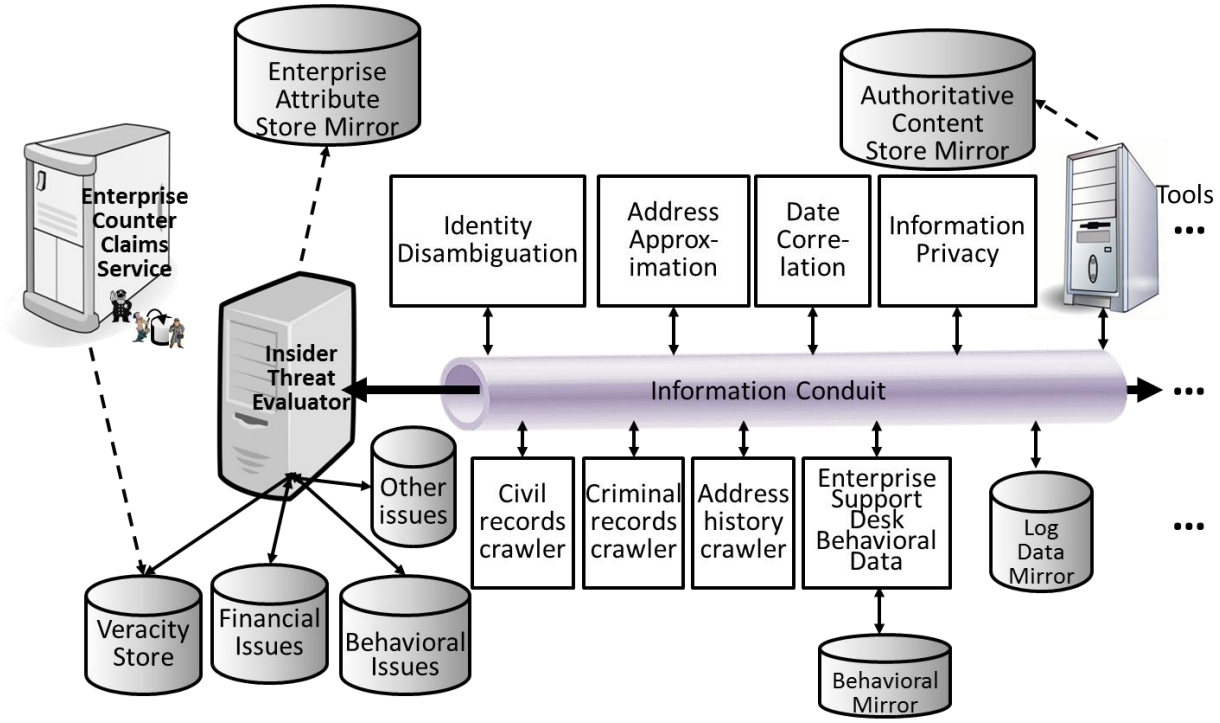

Fig11. Insider Threat System

Person Entities

Person entity factors cover a variety of data about the person and his behaviors and these may come from a variety of sources. These data cannot be considered unless they derive from designated (by the enterprise) authoritative sources. Entity veracity factors are assigned, initially, unit values and may be combined from a number of sources. Unit values may be positive or negative (either increasing or decreasing veracity), and they are applied to veracity measures in a later section. Any previously resolved issues (through vetting or supervisor administrative judgement) may be discarded. Five categories (each with a number of subcategories and each instance is a factor) are delineated below:

Category 1. Community information – characteristics or events that add to the veracity of a person. Each adds a fixed value to overall veracity. Many of these are from sources requiring verification, but some may have sufficient documentation.

a. Ties within the local, regional, and national community. This may also apply to international communities, such as research and academic (positive or negative),
b. Recent job title change. Title changes per se may not be relevant but are a verifying factor for some of the other data (positive or negative),
c. Recent relevant awards or job punishments, these data should have records to support the event(s) (positive or negative),
d. Direct support or doubt from notable entities (Trust transitivity). This trust transitivity such as a trusted co-worker speaking highly or poorly about an individual, should be verified wherever possible (positive or negative).

Category 2. Financial information. Degree of debt or other financial burdens since last vetting. These may be age-and source-sensitive, and they may be attribution-sensitive, as discussed in the next section.

a. Issues with credit cards. Debt and delinquency in credit card accounts may indicated financial problems that would make an individual susceptible to corruption (negative),
b. Large number of credit reports. Usually these indicate shopping for loans even when debt may not reflect these activities (negative),
c. Recent suspicious loan activity. Inaccurate, exaggerated or fraudulent loan applications (negative),
d. Sudden explained or unexplained wealth. Exceptions may be inheritance or legitimate investment windfalls (negative),
e. Debt exceeds ability to pay. This is a strong indicator of financial problems (negative).

Category 3. Legal issues or other stress factors. These may be age-and source-sensitive, and they may be attribution-sensitive, as discussed in the next section.

a. Recent death in family. Even expected deaths may cause a short term stress increase. (negative),
b. Poor job performance rating. Never a good sign and a direct impact to trust (negative),
c. Divorce. Family dis-harmony may cause a large amount of stress, dissatisfaction, or depression (negative),

d. DUI. Drugs and alcohol may be first indicated by a DUI event (negative),
e. Felony or misdemeanor charges. Directly affecting the amount of trust placed in an individual (negative).

Category 4. Discovered secrets. These may be age- and source-sensitive, and they may be attribution-sensitive, as discussed in the next section.

a. Attempts to hide sexual issues. Sexual issues per se may not be an issue, but hiding these may be a source of blackmail by nefarious people trying to co-opt an individual's assistance. (negative),
b. Uncovered alternate identities. Alternate identities are often used for nefarious activities (negative),
c. Residential ambiguity or multiple residences in a locale. Must be confirmed and a rationale established (negative).

Category 5. Unusual behavior. These will generally be from the Enterprise Support Desk Records and may be considered authoritative.

a. Non-cleared travel. Individuals with clearances are expected to report foreign travel. Lack of this reporting may indicated nefarious activities (negative);
b. Unusual and unexplained IT usage (negative),
    i. Unusual downloads (negative),
    ii. Unusual hours of usage (negative),
    iii. Many open applications at same time (negative);
    iv. Logged into more than one computer at the same time, or multiple accounts on the same computer (negative)
c. Sharing of credentials. This is especially worrisome in the ELS application security model where unique identification of individuals and full accountability for action are requirements (negative);
d. Frequent use of backup methods. Indicator, backups indicate a fear of IT corruption or collapse, or the may just be prudent computing usage (negative);
e. Unusual delegations. This is especially worrisome in the ELS application security model where several forms of formal delegation are available to maintain identity and full accountability for action are requirements (negative);
f. Extended on-line absence followed by high activity. Not counting the occasional extended vacation or other rationally explained activity (negative);
g. Unusual hours or time on-line. A different pattern than recorded for an individual. Each person has developed work habits and if they are effective, this should not create a problem. But switching to late hours suddenly may be an indicator (negative).

Non-Person Entities

These factors will generally be from the Enterprise Support Desk Records and may be considered as authoritative. All are negative.

Category 6. Non-Person Veracity

a. Recent attacks. These are considered unless forensics find the vulnerability, the data owner closes the vulnerability, and complete teardown and rebuild has happened since the attacks.
b. Recognized misuse of privilege. Often machine-to-machine privilege is identity based and not carefully monitored. Moving data to other machines and/or acting as a third party proxy are examples of privilege abuse. This may be documented through the enterprise support desk analysis of monitoring data.
c. The host server is physically moved outside (or into) a protected area without a change in enterprise registration. All enterprise assets are registered, and the registration must be updated when any changes occur. How these are discovered is often problematical.
d. Call-out to unknown URLs. This is a known sign of exploitation, and unless the device is being used in counter-cybersecurity, it should be considered for a complete teardown and rebuild. Of course, URL may appear to be unknown when analyzing monitoring records and rationale should be sought.
e. Missing log records. This is a clear sign of nefarious activity or sloppy configuration control.
f. Lenient access and privilege requirements. Privileges granted to the device may be greater than the device uses for its own access. This situation may lead to item b. above.
g. Available software interfaces that are not authorized. One clear step with the ELS application security model is to close all interfaces not being used and remove the software behind those interfaces where possible.
h. Non-uniform identity requirements on interfaces. All interfaces in use should have the same identity assurance requirements or at least identity assurance levels.
i. Missing current patches that are authorized. One example is Industrial Control Systems (ICS) not being patched until they have to be taken off-line. This practice can perpetuate vulnerabilities and invite nefarious activity.

V. CREATING A MODEL & COUNTER-CLAIMS

A simplified model is developed as a start. While weightings may be applied to the various values of data and information veracity factors, it is best to await some actual experience with the representation before beginning that modification. In the previous section, we delineated five basic categories of veracity for person users and a single category for non-person users for evaluation, subject to data sources and correlation. Accordingly veracity is described as an n-tuple shown below:

For Persons:

Veracity = (Community = V1, Financial = V2, Legal = V3, Discovered Secrets = V4, and Behavior = V5) (eq. 1)

For Non-Persons:

Veracity = V6 (eq. 2)

Further, each value, V*i*, has a default value of 1.0 which is appreciated by ΔV for each of the unique factors in each category. For example, using category 1:

$(\Delta V1)_k$ = (+/-0.1) * source factor1 * source factor 2 (eq.3)

For every unique occurrence, *k*, of a factor in paragraph marked category 1.

The default value of 1.0 is reduced by ΔV for each of the unique factors in categories 2–6 where applicable.

$(\Delta Vi)_k$ = (+/-0.1) * source factor1 * source factor 2 (eq.4)

Where *i* = 2-6 for every unique occurrence, *k*, of each subcategory in category 2-6.

Source factor1 is 0.5 for publicly derived data, and 0.25 for publicly derived data without source citation or date of item. Source factor1 is 1.0 for authoritative source data. Source factor2 is 0.5 where attribution is approximate and 1.0 where attribution is certain.

$Vi = 1.0 + \sum_k (\Delta Vi)_k$ (eq.5)

Counter claims will be provided when requested by the data owner in the registration of his/her service. The counter claims will be given as a vector of values:

Counter Claim for a person = (V1, V2, V3, V4, V5, none) (eq.6)

Counter Claim for a non-person = (none, none, none, none, none, V6) (eq.7)

Supervisors and data owners will have claims for access to component data from the insider threat server for subordinates (in the case of supervisors) and for application and service users (in the case of data owners). Issues may be marked as resolved at the supervisor's discretion (subject to attribution and logging). An example would be at periodic vetting, the supervisor may mark some issues resolved.

Actions possible:

1. Threshold for denial of access to resources. Not recommended.
2. Threshold for notification to supervisors and data owners (Recommended).
3. Reduce privilege. Not recommended. This may affect performance reviews and cause the value of veracity to further decline in a self-generated spiral.
4. Upon notification, set up a counseling session with the individual or the owner of the asset to review the issues and seek corrections (Recommended).
5. After review, the data may be manually reset, if desirable, by providing rationale and obtaining appropriate authority.

In all cases, when requested by the data owner, the counter claim will be passed in the SAML.

VI. SUMMARY

The formulation of entity veracity provides a method to monitor insider threats, which is required by presidential directive for some but desirable by all organizations. Certain findings are appropriate at this point:

1. For persons, the data associated with information generated prior to the last formal vetting of the person may be marked as resolved at the supervisor's discretion.
2. For persons, it is not felt that automated responses are warranted at this time.
3. For persons, manual resolutions of unfavorable veracities should be implemented at this time.
4. For non-persons, automated responses may be appropriate.
5. Thresholds and responses should be worked out over time with experience.
6. Self-assessment – data as required by executive order 13587 should be summarized and reported.

The next step is a trial instantiation and the working of the unique security issues discussed in the introductory section of this paper as well as the ethical and legal issues discussed in section 5. The veracity measures can provide a management view into the insider threat and can be used to satisfy the requirement for self-assessment. This work is part of a body of work for high-assurance enterprise computing using web services. Elements of this work are described in [31-40].

ACKNOWLEDGMENT

The authors would like to thank Dr. Margaret Myers for her unflagging support in these efforts. We would also like to thank Lynne Russillo, our technical editor who helps us make sure we say what we mean to say in proper English. Finally we thank Joyce Walker and Paula Giffey for their assistance and administrative support.

REFERENCES

[1] William R. Simpson, and Kevin E. Foltz, "Enterprise Level Security: Insider Threat Counter-Claims," Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering and Computer Science 2017, 25-27 October, 2017, San Francisco, USA, pp112-117

[2] Bill Gertz, "The Cyber Threat: Snowden—Ultimate Insider Threat Missed by NSA Security," September 20, 2016. http://freebeacon.com/national-security/cyber-threat-snowden-insider-threat-at-nsa/, accessed April 17, 2017.

[3] Steve Fishman, New York Magazine, "Bradley Manning's Army of One," July 3, 2011. http://nymag.com/news/features/bradley-manning-2011-7/, accessed April 17, 2017.

[4] Ryan Francis, CSO online, "9 employee insiders who breached security," October 6, 2014.http://www.csoonline.com/article/2692072/data-protection/data-protection-165097-disgruntled-employees-lash-out.html, accessed Apr 17, 2017.

[5] Wikipedia, "Insider threat," https://en.wikipedia.org/wiki/Insider_threat, October 2016.

[6] Barack Obama, "Executive Order 13587 – Structural Reforms to Improve the Security of Classified Networks and the Responsible Sharing and Safeguarding of Classified Information," October 7, 2011.

[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] Ann Margaret Strosaker, Michael Thomas Strosaker, Patent, "Determining veracity of data in a repository using a semantic network," US 8108410 B2, International Business Machines Corporation, January 31, 2012. https://www.google.com/patents/US8108410, accessed Apr 17, 2017

[9] Geoffrey Lee, Patent, "Candidate-initiated background check and verification," US 20050055231 A1, published March 10, 2005. http://www.google.com/patents/US20050055231

[10] Eileen Shapiro, Steven Mintz, Patent, "System and method for providing access to verified personal background data," US 20040168080 A1, August 26, 2004. http://www.google.com/patents/US20040168080

[11] Yu Zhao, Jianqiang Li, Patent, "Hierarchy extraction from the websites," US 20090327338 A1, Nec (China) Co., Limited, December 31, 2009. https://www.google.com/patents/US20090327338

[12] J. Hunker, C.W. Probst, "Insiders and insider threats – an overview of definitions and mitigation techniques," Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA), 2 (2011), pp. 4–27.

[13] Wikipedia, "Integrity",https://en.wikipedia.org/wiki/Integrity, accessed November 2016.

[14] Dictionary.com, http://www.dictionary.com/browse/reputation, "reputation," accessed November 2016.

[15] Gerrit J. van der Geest and Carmen de Ruijter Korver, Microsoft, The Architecture Journal, "Managing Identity Trust for Access Control," July 2008. https://blogs.msdn.microsoft.com/nickmac/2009/05/21/the-architecture-journal/,.

[16] Merriam-Webster, "Veracity," http://www.merriam-webster.com/dictionary/veracity, accessed November 2016.

[17] William R. Simpson and Kevin 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, pp. 56-61.

[18] Office of the Secretary of The Air Force, Office of the Chief Technology Officer (SAF/CTO), Consolidated Enterprise IT Baseline, Version 5, available at https://intelshare.intelink.gov/sites/afceit/TB/_layouts/15/viewlsts.aspx?BaseType=1, not available to all.

[19] Email from Rudy Rihani, Project Manager, Accenture Corporation, dated March 6, 2016, Subject: " manpower savings with ELS"

[20] 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, Feb 2000

[21] 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

[22] 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

[23] 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.

[24] J. W. Butts, R. F. Mills, and R. O. Baldwin, "Developing an insider threat model using functional decomposition," in Computer Network Security, ser. Lecture Notes in Computer Science, V. Gorodetsky, I. Kotenko, and V. Skormin, Eds. Springer Berlin / Heidelberg, 2005, vol. 3685, pp. 412–417. [Online]. Available: http://dx.doi.org/10.1007/11560326 32

[25] R. Chinchani, A. Iyer, H. Q. Ngo, and S. Upadhyaya, "Towards a theory of insider threat assessment," in Proc. of the 2005 International Conference on Dependable Systems and Networks (DSN'05), Yokohama, Japan. IEEE, June–July 2005, pp. 108–117.

[26] Digital Subscriber Line, accessed 9/1/2015. https://en.wikipedia.org/wiki/Digital_subscriber_line

[27] RFC 2516 A Method for Transmitting PPP Over Ethernet (PPPoE), February 1999. http://tools.ietf.org/html/rfc2516

[28] The Channel Co., CRN Magazine, CRN Staff, "How to Avoid the Five Biggest BYOD Mistakes", available at:

https://www.crn.com/blogs-op-ed/channel-voices/240006736/how-to-avoid-the-five-biggest-byod-mistakes.htm, accessed on 26 June 2018

[29] Long, William, Computer Weekly, "BYOD: data protection and information security issues", https://www.computerweekly.com/opinion/BYOD-data-protection-and-information-security-issues, accessed on 26 June 2018

[30] South Carolina Enterprise Information System (SCEIS), "SCEIS Data Cleansing General Guidelines", http://sceis.sc.gov/documents/data_cleansing_guidelines_v2.doc, accessed on 26 June 2018

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

[32] William R. Simpson, and Kevin E. Foltz, "Assured Identity for Enterprise Level Security,"Lecture Notes in Engineering and Computer Science, Proceedings of the World Congress on Engineering 1017, 5-7 July 2017, London, U.K., pp. 440–445,`

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

[34] William R. Simpson, and Kevin E. Foltz, "Enterprise Level Security: Insider Threat Counter-Claims," Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering and Computer Science 2017, 25–27 October, 2017, San Francisco, USA, pp. 112–117.

[35] 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," Los Angeles, CA. September 2017, pp. TBD.

[36] William R. Simpson and Kevin E. Foltz, Proceedings of the 19th International Conference on Enterprise Information Systems (ICEIS 2017), Volume 1, pp. 177–184, Porto, Portugal, 25–30 April, 2017, "Enterprise Level Security with Homomorphic Encryption," SCITEPRESS – Science and Technology Publications.

[37] Kevin Foltz, and William R Simpson, "Enterprise Considerations for Ports and Protocols," Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering and Computer Science 2016, 19–21 October, 2016, San Francisco, USA, pp.124–129.

[38] Kevin E. Foltz, and William R Simpson, "Simplified Key Management for Digital Access Control of Information Objects," Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering 2016, 29 June–1 July, 2016, London, U.K., pp. 413–418.

[39] Kevin E. Foltz and William R. Simpson, 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, pp. 56–61.

[40] Kevin E. Foltz and William R. Simpson, Wessex Institute, Proceedings of the International Conference on Big Data, BIG DATA 2016, "Access and Privilege in Secure Big Data Analysis," 3–5 May 2016, Alicante, Spain, pp. 193–205.