INSTITUTE FOR DEFENSE ANALYSES

Incorporating IoT in Enterprises with ELS

Kevin E. Foltz, *Project Leader*
William R. Simpson

March 2020

IDA Non-Standard
NS D-13134

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

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

About This Publication

This work was conducted by the IDA Systems and Analyses Center under contract HQ0034-14-D-0001, Project BC-5-2283, "Architecture, Design of Services for Air Force Wide Distributed Systems," for the USAF HQ USAF SAF/CIO A6. The views, opinions, and findings should not be construed as representing the official position of either the Department of Defense or the sponsoring organization.

Acknowledgements

For More Information

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

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

Copyright Notice

Rigorous Analysis | Trusted Expertise | Service to the Nation

INSTITUTE FOR DEFENSE ANALYSES

IDA Non-Standard NS D-13134

Incorporating IoT in Enterprises with ELS

Kevin E. Foltz, *Project Leader*

William R. Simpson

Incorporating IoT in Enterprises with ELS

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

Abstract — A number of small computing devices, mechanical and digital machines, objects, sensors, and controllers of other devices are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. These devices are functional and inexpensive, and are termed collectively as the Internet of Things (IoT). Many of the devices do not have enough memory or computing power to participate in even basic security. How does one take advantage of the IoT functionality while maintaining security functionality in high assurance systems? This paper examines a preliminary formulation that is pertinent to the Enterprise Level Security (ELS) model.

Index Terms — Appliance, ELS, Internet of Things, IoT, IT security, Small Devices

I. INTRODUCTION

TOday, there are over 7 billion people on this planet. Connectivity, in the form of internet capability, has been spreading for the last 20 years and now encompasses over 90% of the Earth's surface. Over 1 billion devices have internet protocol (IP) addresses and may be on the internet at any time. Of these devices, the vast majority are small and inexpensive carriers of information that fall into the category of the Internet of Things (IoT) [1].

Internet capability has become more widely available, and through provider services, broadband, and other means, the cost of connecting is decreasing. Devices with Wi-Fi, Broadband, IR, and Bluetooth, are developed. They may have multiple sensors and controllers built into them. Technology costs are going down, and smartphone penetration is skyrocketing. All of these things are creating a "perfect storm" for the IoT, as its usage increases exponentially [2].

IoT includes an extraordinary number of objects, ranging from self-driving cars and smart microwaves to wearable fitness devices that measure heart rate and the number of steps taken in a day. There are even connected footballs, toothbrushes, doorbells, and an array of single-point sensors that integrate with other devices to provide overviews and courses of action [3].

However, there is little concern about security. Integrity and confidentiality require computing power, and many of the IoT devices do not have enough resources to provide these. This work is part of a larger body of work termed Consolidated Enterprise IT Baseline (CEITB). In this paper, we review the communication models for web services.

Manuscript received 24 January 2020; revised 11 March 2020. 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).

We then review ELS and its basic architecture. Next, we review the threats considered, including how they affect server configuration and how to configure firewalls for port blocking. Finally, we provide the unique factors that arise with IoT and explain how to handle their properties within this high security environment.

II. ENTERPRISE LEVEL SECURITY

A. *Security Process Background*

Enterprise Level Security (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 is 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 [4]. From there, a set of enterprise level requirements is formulated that conforms to the tenets and any high-level guidance, policies, and requirements.

B. *ELS Framework*

The ELS framework has evolved from a fortress approach, in which the threat is assumed to be stopped at the front door, to a distributed security system that eliminates or mitigates many of the primary vulnerability points inherent in that system, as shown in Figure 1. The basic process of identification involves a two-way contract between two entities that are initiating a communication. Each entity needs to have some assurance that the party they are engaged with is a known entity and, specifically, the one to whom the communication should be allowed. This is done by the presentation of claims by each party that are verifiable and may be validated. These claims are often in the form of credentials. The basic process is described extensively in [5].

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

Current paper-laden access control processes for an enterprise operation are plagued with ineffectiveness and inefficiencies. In a number of enterprises, tens of thousands of personnel transfer locations and duties annually, which introduces delays and security vulnerabilities into their operations on a daily basis. ELS mitigates security risks

while eliminating much of the manual system administration required to grant and remove user/group permissions to specific applications/systems. Early calculations show that 90–95% of recurring person-hours in government and the defense industry are saved and up to 3 weeks in delay for access request processing are eliminated by ELS-enabled applications [6]. Although 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.

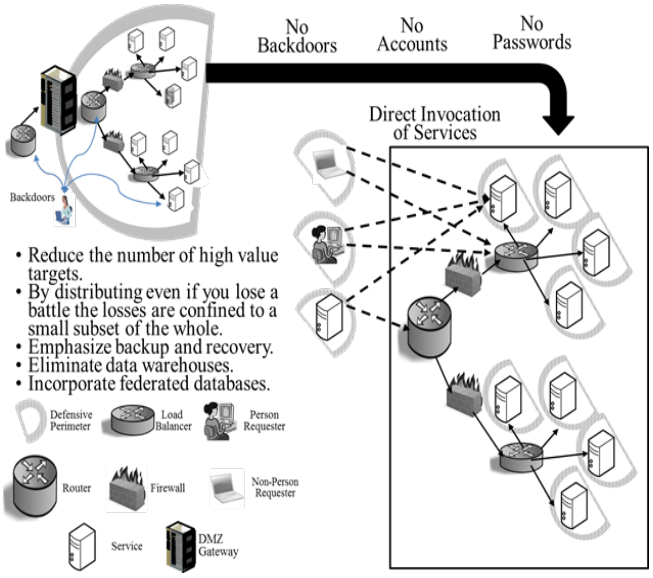

Figure 1 Distributed Security Architecture

C. Security Principles

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

- Know the Players – this is done by enforcing bilateral 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 – this means ensuring the receiving party knows that he received exactly what was sent;
- Require Explicit Accountability – this requires monitoring, logging, and reviewing transactions.

Know the Players

In ELS, the identity certificate is an X.509 public key infrastructure (PKI) certificate [7]. 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 authentication factors (in combination with PKI) may be required from certain entities, such as identity confirming information or biometric data. The authentication is bilateral and it requires both the requester and provider to have PKI certificates. The certificate may reside with the server in the case of the provider, but if the application is a requester of other services, it must also have a PKI certificate. A certificate is required for identity even when the entity is an IoT device.

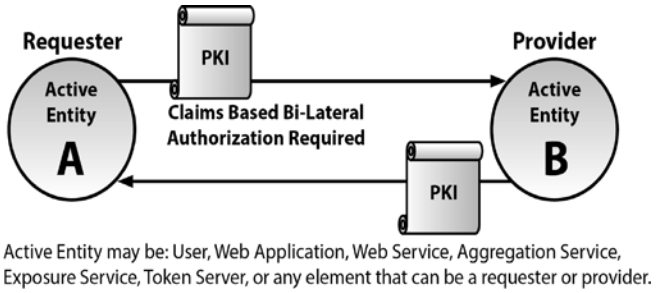

Figure 2 Bilateral Authentication

Maintain Confidentiality

Figure 3 shows that ELS establishes end-to-end transport layer security (TLS) [8] encryption (and never gives away private keys that belong uniquely to the certificate holder). The private keys that belong uniquely to the certificate holder are held in hardware storage. These keys may be present in personal identity verification (PIV) cards with embedded chips for individuals and Hardware Storage Modules (HSMs) 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. Confidentiality is required for identity even when the entity is an IoT device.

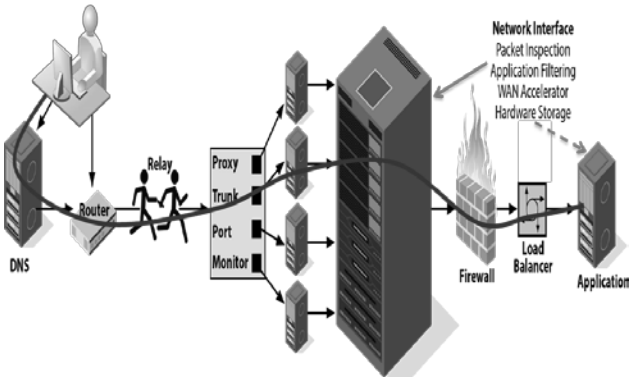

Figure 3 End-to-end Encryption

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 information. As shown in Figure 4, access control credentials utilize the Security Assertion Markup Language (SAML) (SAML authorization tokens differ from the more commonly used single sign-on (SSO) tokens, and are not used for authentication in ELS.) [9]. 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. This separation is required even when the entity is an IoT device.

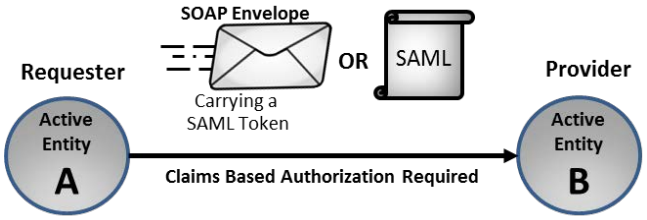

Figure 4 Claims-based Authorization

Maintain Integrity

Integrity implementation is 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 as it is not as strong as end-to-end integrity. At the application layer, packages (SAML tokens etc.) are signed, and signatures are verified and validated [10]. Integrity is required even when one of the entities is an IoT device.

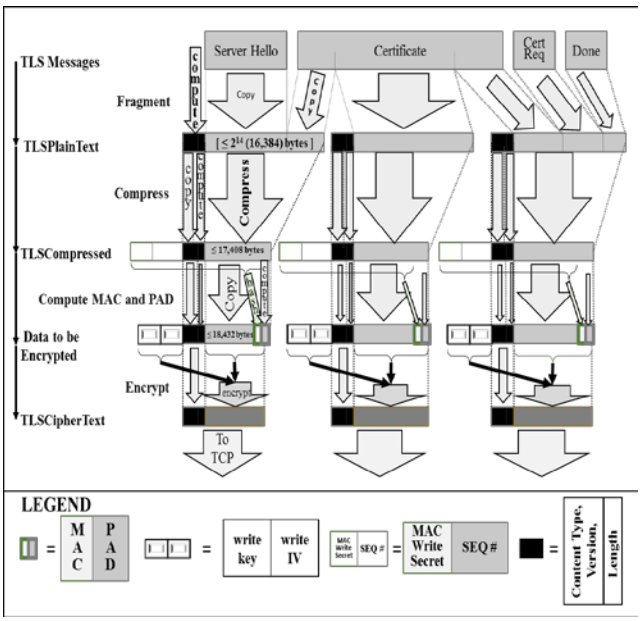

Figure 5 Integrity Measures

Require Explicit Accountability

All active entities with ELS are required to act on their own behalf (no proxies or impersonation allowed). For larger enterprises, a repository is recommended (as shown in Figure 6) to allow ELS services to monitor 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 log records to an enterprise store. In this environment, tools may periodically review the records for nefarious behavior. 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 [11] [12]. Because this activity is recommended for only large enterprises, it is not recommended in the minimal instantiation of ELS. Accountability is required especially when the entity is an IoT device.

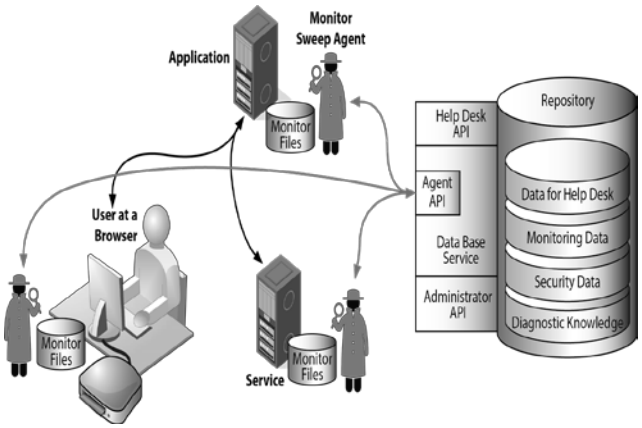

Figure 6 Accountability through Centralized Monitoring

III. ENDPOINT PROTECTION IN ELS

In ELS, an agent-type model is preferred, one in which the packet header filtering and other security functions reside at the web server in the handler chain of the web service. The basic configuration of endpoint protection in ELS is shown in Figure 7 and provides a complete set of security functions for packet, message, and application layer security, tailored for the specific web service being protected. The new functions that are added in the server are packet header inspection, packet content inspection, message content inspection, and application protection. These functions implement the ports and protocols protection, as well as other security functions normally provided by network devices such as intrusion detection/protection, packet and message content filtering, deep packet inspection, and application/web content filtering such as included in an application firewall.

A service requestor establishes communication with the server hosting the target web service according to the ELS practice using HTTPS. The packet is received by the destination server and the packet header is immediately inspected to perform the ports and protocols blocking, source whitelist/ blacklist checking, and other filtering based on only the header, including stateful tracking of client addresses and ports. Until an HTTPS session has been established, only packets addressed to the server's IP address and port 443 are allowed. Other ports may be opened as needed as part of the web service following HTTPS establishment.

On the return path, the messages follow a similar process. In effect, the packet header inspection module can perform the required network-layer filtering and can block traffic based on ports and protocols (protocol, IP address, and port).

In the ELS endpoint protection architecture, the endpoint protection modules can be configured to communicate with additional security monitoring appliances, such as a NetScout, that can compile and track statistics about the security status of the server and the web service. The security appliances should be active entities and communicate with the server via TLS with mutual authentication. If required, the server could send the

decrypted message traffic to other security appliances through this interface for additional security functions.

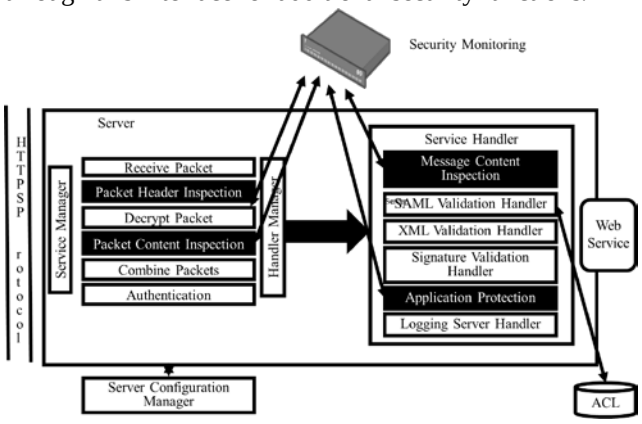

Figure 7 ELS Endpoint Security Functions

The endpoint protection functions are configured through the server configuration management interface, which communicates with the server by TLS with mutual authentication. The ports and protocols, whitelist information, and any software updates are provided through this interface.

It is recommended that the initial configuration of the packet header deny all ports and protocols, both incoming and outgoing (as opposed to the traditional incoming only), and that permissions be configured in when they are identified as needed.

IV. HANDLING AND INSPECTION OF TRAFFIC

Handling and inspection is done in software-only modules in the server. The software functionality is embodied in handlers in the handler chain of the server as shown in Figure 8.

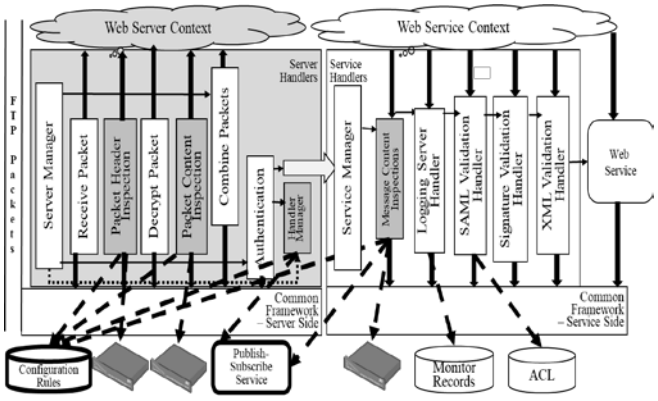

Figure 8 Server Side Handlers

Note that the handlers are embedded in the server handler chain at the point that the communication is prepared for their use and that the functionality has been divided along those lines. These services are now distributed to packet header inspection, packet content inspection, and message content inspection. Each of these may perform inspection related to intrusion detection or blacklist blocking. This is the preferred embodiment for enterprise applications. It moves the inspections to the point of the application itself by inserting handlers within the server and service to do the inspections at the stage where it makes the most sense. The inspections that can be done without decrypting the packets may be done at the front of the web server because entities performing inspections without decryption are considered passive entities. Moving inspections of decrypted traffic inside the server not only preserves the end-to-end paradigm, it encapsulates the security and allows tailoring for the application itself. The encapsulated security with the application is virtualization ready.

Inspections are required of all entities and many of the IoT devices do not have sufficient processing and memory for these inspections.

V. IoT DEVICES

For the class of small special application devices that measure environments and/or control specific hardware or both, special security considerations must be taken. These devices are increasingly becoming the target of attacks from Mirai to WannaCry [13]. The IoT is primarily about functionality. The IOT security domain has not yet matured.

The first step toward IoT security is to field only the IoT devices deemed essential for the enterprise. Such IoT devices will be fronted with a device end-point agent and physical protection. The physical protection encompasses both the IoT device and the agent. It hides the native IoT interfaces and communication channels from the outside and makes them available only to the agent. The agent is then the single external interface for the combined system. The device end-point agent is software on enterprise-approved devices that interacts with central services. The agent can run natively on the IoT device, where such capabilities are provided, or it can be part of a separate hardware device that attaches physically or wirelessly to the IoT device. For ELS purposes, IoT devices are of three types:

1. Individual full system devices with a single capability and an enterprise management system. This IoT is an registered device and it is configured with a secure key store and an end-point agent and registered in the end-point registry as shown in Figure 9. The enterprise end-point service(s) are configured to check this device for registry and attestation and distribute the information to a designated end-point. Such a device resembles a phone, tablet, or laptop, but often has minimal software and special hardware. It is a very capable IoT device, and such devices are rare.

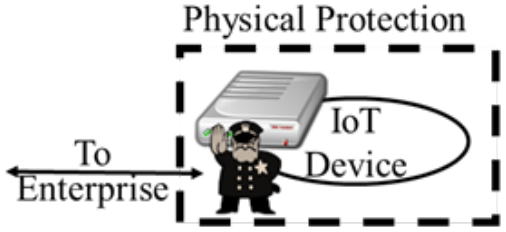

Figure 9 Full System IoT Device

2. Sensors, aggregations of capabilities, and other collections of devices that are part of a network of devices reporting to a single manager of the IoT

collection. The individual elements of the collection are not considered registered devices, but the managers of the IoT collection are enterprise-registered devices, are configured with a tamper-proof secure key store and an end-point agent, and are registered in the end-point registry, as shown in Figure 10. The enterprise end-point service is configured to check this device for registry and attestation and distribute the information to a designated end-point. In this case, the collection manager is the single interface for all devices, and hardware protections must encompass the collection manager, its agent, if applicable, and all of the sensors or other devices in the entire collection. This works best for physically localized collections.

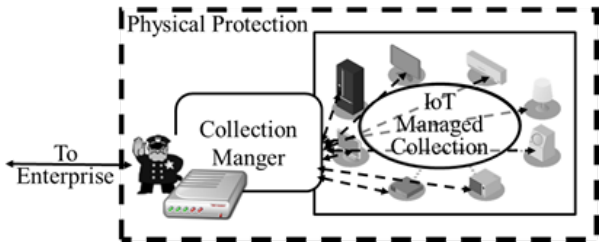

Figure 10 Managed Collection of IoT Devices

3. Individual devices with a single capability but less than full system capability and unable to act as an ELS compliant device. These devices may be employed only when they are hardwired to a security appliance (sometimes called a "bump in the wire") that will provide all of the ELS compliant security (see Figure 11). The device will have only the hardwired interface active, and all other ports will be shut down, including any ports designed to communicate with the manufacturer. All communications will be handled by the security appliance (the appliance may mimic device communications such as Wi-Fi, Broadband, or Bluetooth), and both the device and the security appliance are treated as a single entity from an enterprise standpoint.

Figure 11 Security Appliance Fronted IoT Device

For each of these device types, the key features are the following:

1) ELS communication with the enterprise, either natively or through a special device that mediates the IoT protocols to ELS protocols.
2) Physical protection boundary that includes the IoT devices and the ELS interface.
3) Locked down non-ELS IoT interfaces that communicate only with mediation device.

By providing these features, IoT security is comparable to that of standard ELS endpoints with full security capabilities.

VI. SUMMARY

We have reviewed the ELS security model and the IoT application types within the enterprise. We have also described the issues they raise and the vulnerabilities that may be introduced. For enterprise operations, defining an agent approach means a reduced attack space. We have also reviewed the specific requirements for an enterprise level security that is bilaterally authenticated and encrypted end-to-end. This paper is part of a body of work for high-assurance enterprise computing using web services. Elements of this work are described in [14-30].

REFERENCES

[1] Quora, Future of the Internet, "What exactly is Internet of Things (IoT)?" https://www.quora.com/What-exactly-is-Internet-of-Things-IoT?redirected_qid=7001614, accessed on 24 September 2019.

[2] Jacob Morgan, Contibuter, Forbes.com, "A Simple Explanation Of 'The Internet Of Things'" https://www.forbes.com/sites/jacobmorgan/2014/05/13/simple-explanation-internet-things-that-anyone-can-understand/#21acfe371d09, May 2014, accessed on 24 September 2019.

[3] Jen Clark, IBM.com, "What is the Internet of Things?" https://www.ibm.com/blogs/internet-of-things/what-is-the-iot/, Novemeber 2016, accessed on 24 September 2019.

[4] 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.

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

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

[7] 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

[8] 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

[9] 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

[10] 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.

[11] 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

[12] 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.

[13] Mathew J. Schwartz, "Attacks Targeting IoT Devices and Windows SMB Surge," Bank Info Secuirty, https://www.bankinfosecurity.com/attacks-targeting-iot-devices-windows-smb-surge-a-13082, , last accessed on 10 October, 2019.

[14] William R. Simpson and Kevin E. Foltz, Lecture Notes in Engineering and Computer Science, Proceedings World Congress on Engineering and Computer Science (WCECS) 2018, Volume 1, "Mobile Ad Hoc for Enterprise Level Security," pp. 172-177, Berkeley, CA. October 2018, ISBN: 978-988-14048-1-7, ISSN: 2078-0958.

[15] William R. Simpson and Kevin E. Foltz, Proceedings of The 20th World Multi-Conference on Systemics, Cybernetics and Informatics: WMSCI, "Enterprise Level Security - Basic Security Model", Volume I, WMSCI.

[16] William R. Simpson and Kevin E. Foltz, 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,

[17] William R. Simpson and Kevin E. Foltz,Proceedings of the 21ST International Command and Control Research and Technology Symposium (ICCRTS), "Federation for a Secure Enterprise," http://www.dodccrp-test.org/s/paper_2.pdf, London, UK. September 2016.

[18] William R. Simpson and Kevin E. Foltz, Proceedings of the Information Security Solutions Europe (ISSE) 2016, ISBN: 9781541211445, "The Virtual Application Data Center," pp. 43-59, https://www.amazon.com/isse2016-3-Information-Security-Solutions-Europe/dp/1541211448, Paris, France, November 2016.

[19] William R. Simpson and Kevin E. Foltz, Haeng Kon Kim • Mahyar A. Amouzegar (eds.), Transactions on Engineering Technologies, Special Issue of the World Congress on Engineering 2015, Chapter 15, pp. 205-220, "High Assurance Asynchronous Messaging Methods," 15 pp., DOI 10.1007/978-981-10-2717-8, Springer Dordrecht 2017.

[20] William R. Simpson and Kevin E. Foltz, Lecture Notes in Engineering and Computer Science, Proceedings of the World Congress on Engineering (WCE) 2017, "Assured Identity for Enterprise Level Security," pp. 440-445, Imperial College, London, July 2017, IBSN: 978-988-14047-4-9.

[21] William R. Simpson and Kevin E. Foltz, Proceedings of The 21th World Multi-Conference on Systemics, Cybernetics and Informatics: WMSCI, "Data Mediation with Enterprise Level Security," WMSCI 2017, Orlando, Florida, 8-11 July 2017, 6 pages.

[22] William R. Simpson and Kevin E. Foltz, Proceedings of the 22nd International Command and Control Research and Technology Symposium (ICCRTS), "Escalation of Access and Privilege with Enterprise Level Security," ISBN: 978-0-9997246-0-6, Los Angeles, CA. September 2017.

[23] William R. Simpson and Kevin E. Foltz, Lecture Notes in Engineering and Computer Science, Proceedings World Congress on Engineering and Computer Science (WCECS) 2017, Volume 1, "Enterprise Level Security: Insider Threat Counter-Claims," pp. 112-117, Berkeley, CA. October 2017.

[24] William R. Simpson and Kevin E. Foltz, Sio-Long Ao, et. al. (eds.), IAENG Transactions on Engineering Sciences, Special Issue of the Association of Engineers Conferences 2016, Volume II, pp. 475-488, "Electronic Record Key Management for Digital Rights Management," 14 pp., World Scientific Publishing, Singapore, ISBN 978-981-3230-76-7, 2018.

[25] William R. Simpson and Kevin E. Foltz, "Secure Identity for Enterprises," IAENG International Journal of Computer Science, vol. 45, no. 1, pp. 142-152, ISSN: 1819-656X, February 2018.

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

[27] William R. Simpson and Kevin E. Foltz, Proceedings of the 8th International Conference on Electronics, Communications and Networks (CECNet 2018), Volume 1, "Cloud Security and Scalability," p. 27, Bangkok, Thailand, November 2018.

[28] William R. Simpson and Kevin E. Foltz, "Insider Threat Metrics in Enterprise Level security," IAENG International Journal of Computer Science, vol. 45, no. 4, pp. 610-622, ISSN: 1819-656X, December 2018.

[29] Simpson W. and Foltz K., Lecture Notes in Engineering and Computer Science, Proceedings World Congress on Engineering and Computer Science 2015, Volume 1, "Maintaining High Assurance in Asynchronous Messaging," pp. 178–183, Berkeley, CA, October 2015.

[30] William R Simpson, and Kevin E. Foltz, "Mobile Ad-hoc for Enterprise Level Security," Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering and Computer Science 2018, 23-25 October, 2018, San Francisco, USA, pp. 172-177.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE (DD-MM-YY)	2. REPORT TYPE	3. DATES COVERED (From – To)
00-03-20	Non-Standard	

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Incorporating IoT in Enterprises with ELS	HQ0034-14-D-0001
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBERS

6. AUTHOR(S)	5d. PROJECT NUMBER
William R. Simpson, Kevin E. Foltz	BC-5-2283
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESSES	8. PERFORMING ORGANIZATION REPORT NUMBER
Institute for Defense Analyses 4850 Mark Center Drive Alexandria, VA 22311-1882	NS D-13134

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR'S / MONITOR'S ACRONYM
Frank P. Konieczny United States Air Force SAF-CIO A6, 1800 Air Force Pentagon, Washington DC 20330-0001	USAF HQ
	11. SPONSOR'S / MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

Project Leader: Kevin E. Foltz

14. ABSTRACT

A number of small computing devices, mechanical and digital machines, objects, sensors, and controllers of other devices are provided with unique identifiers (UIDs) and the ability to transfer data over a network without requiring human-to-human or human-to-computer interaction. These devices are functional and inexpensive and are termed collectively as the Internet of Things (IoT). Many of the devices do not have enough memory or computing power to participate in even basic security. How does one take advantage of the IoT functionality while maintaining security functionality in high assurance systems? This paper examines a preliminary formulation that is pertinent to the Enterprise Level Security (ELS) model.

15. SUBJECT TERMS

Appliance, ELS, Internet of Things, IoT, IT security, Small Devices

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Frank P. Konieczny
Unclassified	Unclassified	Unclassified	Unlimited	7	19b. TELEPHONE NUMBER (Include Area Code) 703-697-1308