Enterprise End-point Device Management

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

***Abstract* — Enterprise systems have traditionally managed network security with firewalls, virtual private networks (VPNs), antivirus software, and computers imaged and deployed from within the enterprise system. This implies a fortress model, in which a clear boundary lies between what is inside the fortress and what is outside. Those assets inside are protected from the outside. This model does not match the current world. Mobile devices, which are outside the traditional fortress, are now a part of everyday life and thus a part of everyday business. Such devices are not add-ons to a managed core but instead are part of the core of the enterprise. A modern enterprise depends on collaboration and communication across devices regardless of platform, and security requires all devices to be registered and managed with mobility in mind. End-point device management is the process by which enterprise hardware and the software that runs on it are managed, updated, validated, and approved by the enterprise. Information about users, including which end-point device they are logged into, their geo-location, and other factors, are maintained within the enterprise. These are made available for maintenance and update, as well as access and privilege determination, restriction, and elimination. Rogue and compromised devices (and individuals) are removed from the registry and prevented from interacting with enterprise services. This enables the enterprise to tightly control the security properties of not just the connections to the services but also the end-point devices where these communications originate.**

Index Terms* — *Device Management, End-point, Enterprise Level Security, High Assurance, Mobile Devices

I. Introduction

The current model of device security is based upon a fortress approach with well-defended entry points. When mobile devices began to proliferate, and in forms that were unanticipated, it became apparent that a separate management system was needed to secure the multitude of devices that were not under control of the computing center. Within the computing center a legion of administrators maintained servers, keeping them updated, patched, and in proper configuration, but the mobile devices were not always on and connected and often nowhere near the administrators of the computing system. Several designs for Mobile Device Management (MDM) were provided [1-3], and – many of these included provisions for devices provided by the enterprise members, known as Bring Your Own Device (BYOD) [4].

Manuscript received 1 February 2018; revised 15 March 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.
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 fortress model – hard on the outside, soft on the inside – assumes that the boundary can prevent all types of penetration [5], but this assumption has been proven wrong by a multitude of reported network-related incidents. Network attacks are pervasive, and nefarious code is present even in the face of system sweeps to discover and clean readily apparent malware.

Enterprise Level Security (ELS) is a distributed capability designed to counter adversarial threats by protecting applications and data with a dynamic claims-based access control (CBAC) solution. Many of today's enterprise solutions involve a combination of devices that are located within the computing center or elsewhere, making the distinction of mobile devices somewhat blurred. An aircraft may have several servers running onboard inflight, and a command post set up for a temporary period may also have such an array of capabilities. Users may access these from an office, at home, in a partner's facility, or on the road. ELS helps provide a distributed high-assurance environment in which information can be generated, exchanged, processed, and used.

This paper discusses device management within ELS as an end-point management problem. Devices and end-points within the computer center are managed by the same processes used for mobile devices, reducing the need for administrator actions. This includes mobile and non-mobile devices, as well as any device that can be an end-point within the enterprise. 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 [6]. From there, a set of enterprise-level requirements is formulated that conforms to the tenets and any high-level guidance, policies, and requirements.

II. Enterprise-Registered Device Requirements

Some devices inside the enterprise are directly within physical boundaries that are controlled by enterprise personnel. They include devices that host servers for web applications and web services, utility devices to host network monitoring, load balancers, routers, and domain name service resolvers. These devices are fully in the control of the enterprise. The hardware, software, and networking are all enterprise-owned and enterprise-registered.

With increasing computation power in smaller devices, many of the functions traditionally implemented on fixed-location devices are now hosted on mobile devices. For simplicity and consistency, all active entities use enterprise-registered devices to access or provide secure services within the enterprise. This includes servers, desktops, laptops, tablets, phones, watches, network appliances, and any other computation device capable of web interactions within the enterprise. These types of devices are enterprise-

ISBN: 978-988-14047-9-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

registered regardless of whether or not they are mobile. It is impossible to determine whether an end-point is mobile based on its function, so all functions and devices are assumed to be mobile unless registered as fixed enterprise assets confined to an enterprise computing center, such as the devices hosting back-office services and managed accordingly.

The primary requirement for enterprise-registered devices is to be enterprise-approved hardware and contain a tamper-proof method (preferably hardware) for Secure Key Storage and Use (SKSU) with attestation. One such standard for this function is the Trusted Platform Module (TPM) [7]. SKSU is the starting point of trust for enterprise-registered devices. The SKSU manages a public/private key pair, the private key of which cannot be removed or copied from the SKSU. The public key is recorded in the device registry when the device is issued to a user. All future communications with the device are tied back to this key pair. The device proves ownership of the private key in order to provide validated information about the device and its properties, such as installed or connected hardware, installed operating system, installed software, and configuration settings. The SKSU is integrated into the operating system in order to properly account for application and configuration changes. The SKSU is implemented at a sufficiently low level to prevent software attempts to subvert it. This is necessary in particular to prevent leakage of the private key. The SKSU on a mobile device has provisions for storage of derived PKI certificates for authorized users and temporary certificates for guest users [8].

In order to properly use the SKSU for management functions, a software agent is installed on the device, which communicates with enterprise services to establish secure connections and provide proof that the device is in compliance with enterprise security rules and settings. Without communication from the agent, the claims-based process is interrupted and access to enterprise services is denied. The agent itself does not provide security functions, and it is not a trusted end-point, so it could be compromised without harm to the enterprise. It is installed initially by the enterprise, and it is considered an untrusted agent that provides potentially trusted information (i.e., a passive entity). It is simply a functional unit to provide SKSU information and other verifiable information from the device to the enterprise services using the proper formats and protocols. The agent itself can be validated by sending a SKSU-signed attestation of the software on the device. The agent thus asserts its validity through the SKSU as a trusted agent.

Enterprise-registered devices are enterprise working devices and not for personal use. Download of applications is restricted by the enterprise to approved applications, and such software is maintained by the enterprise. Although a browser is provided, it is for communication with the enterprise and is white-list controlled. The end-point device can be disabled by the end-point device manager for any number of reasons including suspicious history, corruption of the software set as indicated by the device attestation report, or improper use.

III. Device Registry

The enterprise device end-point registry consists of a data base of devices, serial numbers, properties, machine certs/keys, locations, attestation reports, who (persons and/or organizations) the device was issued to, who (persons and/or organizations) the notifications are sent to and how, whether the device is Personal Identification Verification (PIV) card-enabled for registered users for the device (for mobile this includes those that have derived credentials recorded with the SKSU) and whether or not a guest logon is allowed, software update status, and incident report reference for the device (if any) and other pertinent data. This database is used by the data registration service, the software update service and other services within the enterprise. Registration and configuration of server end-points, including end-point agents, is done through the data owner and the end-point device registrar. The end-point registrar is an AF-approved and -trained individual who is assigned to a registration unit (similar to PIV issuance stations). Each base is assumed to have at least one registrar. Mobile devices will be registered and configured by enterprise support, including the loading of the current containers, end-point agents, and other software for the particular device, as well as creation and loading of the derived credentials necessary into the SKSU. The standard configurations are available to the registrar in a database. This database is part of or linked to the end-point authorized software and updates. This will be redone on a periodic basis when derived credentials need to be redone (usually with expiring PIVs (if planned to coincide with PIV renewals, it would minimize administrative time)). All other devices will be configured and registered by the administrator assigned to the device in communication with the device registrar. This will minimize updates needed during usage.

The elements of registration are provided in Figure 1. The device registration service captures input from the device and the device registrar. The latter includes any information about the device that requires manual entry. The device registrar authorizes the AF standard software for the device. This may be in a separate store or as part of the software updates store. If this is a renewal, the software will be updated to the latest configured and approved software states. The registration registrar also creates and stores derived credentials based upon the PIV issued by the authorized Certificate Authority (CA) for the device. The registration service stores these data in the device end-point registry. The registration service communicates with the device end-point agent in order to confirm that the attestation report from the agent satisfies what is stored in the device end-point registry. The device end-point registry stores the latest valid attestation report for each device, as well as a history of such reports that shows changes over time as appropriate. Any unauthorized change in attestation reports signals a security alert for that device and possible remediation actions.

The device end-point agent establishes communication with the enterprise end-point service when connectivity is available to the device. Thereafter, the agent provides a heartbeat [9] at a configurable interval, which begins at device connection and periodically sends the IP address,

ISBN: 978-988-14047-9-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

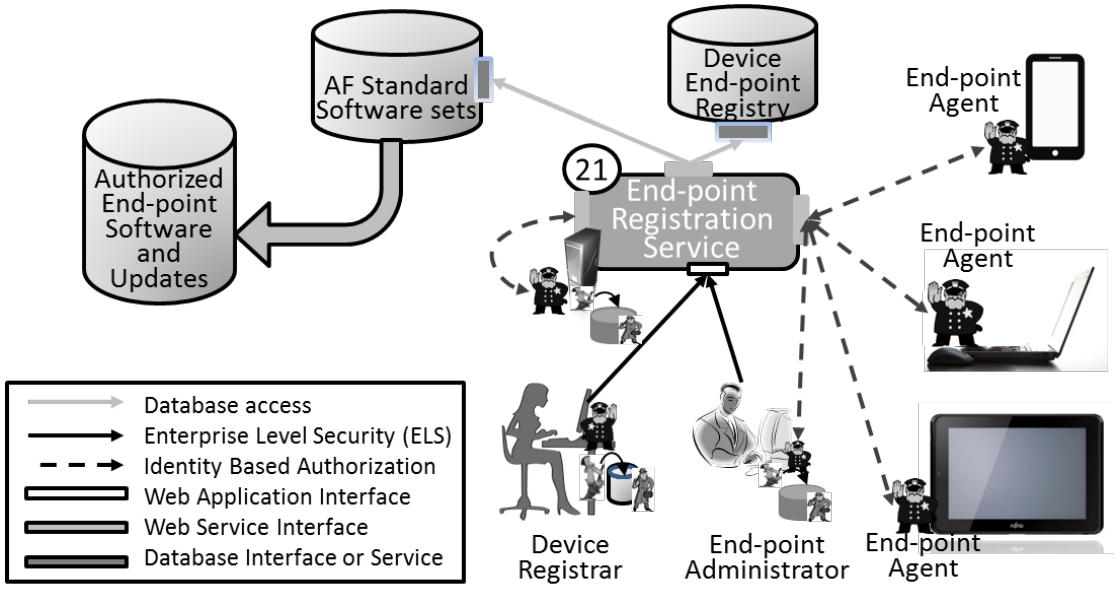

Figure 1: Elements of Registration

device ID, attestation, attestation state, location, and other information. Information unchanged since the last heartbeat is omitted. When the user signs on, the user Distinguished Name (DN) and credential type are added. The periodic reporting by an agent was first described by Hong, et al. in 2001 [10]. As described in [10], most modern network devices are equipped with management agents, typically an Simple Network Management Protocol (SNMP) agent [11] for computer network devices and a Common Management Information Protocol (CMIP) agent [12] for telecommunication network devices. What makes this management approach different is the use of on-device attestation. These data are placed in the end-point/user dynamic binding store. Missed heartbeat cycles (configurable number), for whatever reason, result in the entry being dropped from the end-point/user dynamic binding. The heartbeat is re-established after connectivity is restored. The agent also provides mandatory log files of activity for the Mobile Device Mandatory Log Files Store, which is periodically swept by the monitor sweep agents. Heartbeats may be of configurable durations. Servers in fixed locations and expected to be active may have a less frequent heartbeat than mobile devices that are subject to more dynamic data.

The device end-point agent establishes communication with the enterprise end-point service (shown in Figure 2) when a user logs on to an end-point device. The purpose is to provide a dynamic binding between the user and the device (including such other information as location) for use by other enterprise services and particularly the Provide Claims service (denoted as service 11), which could have restrictions for devices, locations, and other uses of designated enterprise services. Device data is periodically renewed through the heartbeat mechanism described above. When a user logs off of the device, the name is deleted from the dynamic binding store, and logs are provided to the mobile device activity log stores. The heartbeat continues until connectivity with the end-point service is lost.

The Enterprise end-point service (designated 22) communicates with the end-point update manager (designated 23), which stores enterprise-approved software and updates in the Authorized End-point Software and Updates store. When an update is available for enterprise end-point devices, the end-point administrator provides the update to the End-Point Update Manger, which places the update in the end-point software, updates the store, and scans the registration service for candidates for the update, annotating the device end-point registry as appropriate. The software update has a completion required date, and notice is provided to the recorded notifying individuals. Those individual log onto the end-point device and pull the update from the Enterprise End-point Service through the end-point agent. If this is not accomplished by the required date, the end-point update manager notifies the relevant individuals and pushes the update to the device when it is logged onto the system.

The enterprise end-point service:

1. Manages the end-point/user dynamic binding store. The enterprise end-point service verifies end-point presence in the device end-point registry and the equivalence of the SKSU registry information (refuses connection where these fail), stores data provided by the end-point agents when checks are successful, and answers queries from the provide claims web service.
2. Stores the mobile device mandatory log files when these are provided by the end-point agents.
3. Relays end-point agent instructions for notification and/or updates as provided by the end-point updates manager (described in the next section).
4. Issues reboot, shut down, credential revocation, and other termination activities when its own analysis indicates the need or when directed to do so by an authorized entity.
5. Receives and distributes Internet of Things (IoT) data as configured.

Software patches are initially made available to the devices by notification to the registered device users, who initiate an update through the device end-point agent or other designated means. Patches that have not been updated in a reasonable time or whose updating is urgently required

ISBN: 978-988-14047-9-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

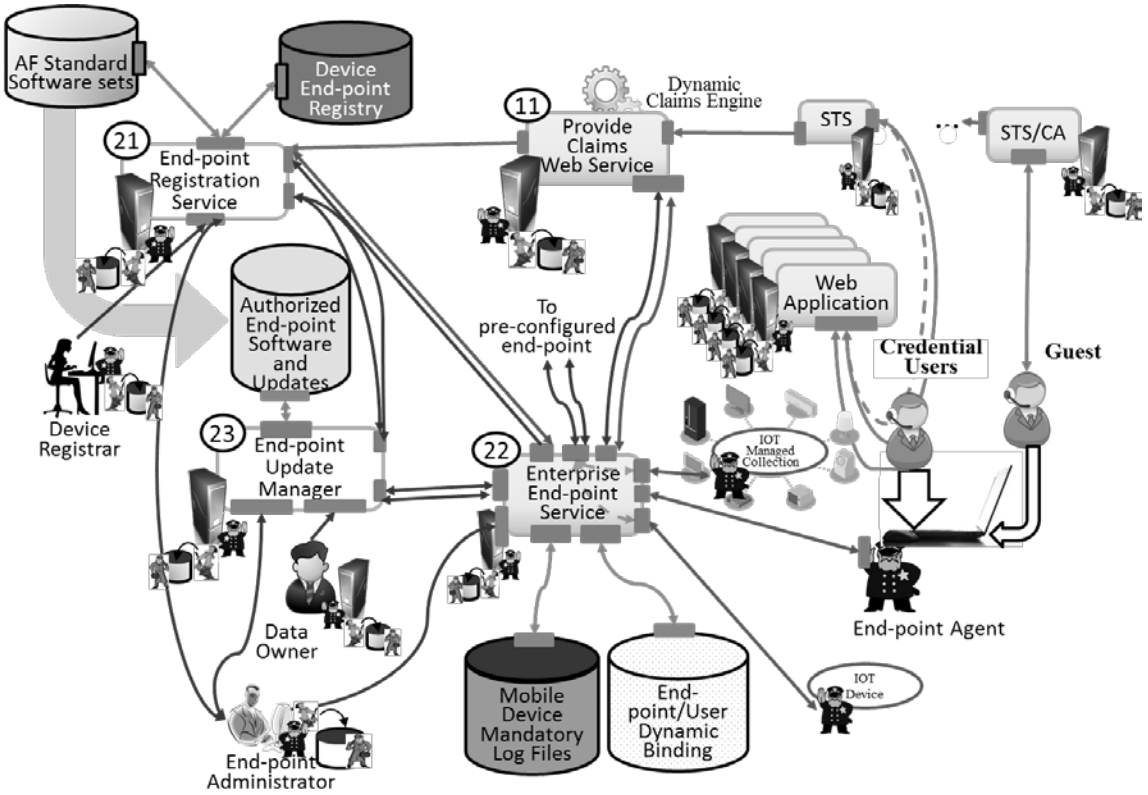

Figure 2 End-point Device Management Process

are pushed to the device. In both cases, the device registration is updated through the device end-point agent, which provides a revised attestation report to the registration service.

Changes in applications are registered and approved. New attestations of approved updates are reregistered in the central store.

The end-point update service contains a schedule for updates that includes availability dates, notification dates, and required completion or push dates. The notifications are provided for the relative end-point, based upon the software installed in the device end-point registry to the individual(s) of record in the device end-point registry by the method(s) indicated in the device end-point registry.

The calendar is used for notifications of scheduled and unscheduled outages as provided by the data owner of the scheduled and unscheduled outage. The notifications are provided for the relative end-point, based upon the software installed in the device end-point registry to the individual(s) of record in the end-point registry by the method(s) indicated in the device end-point registry.

An end-point is terminated when a serious deviation from policy is detected, attestation has not been maintained, where the device is suspected to be compromised, or when the device has been involved in nefarious behavior. This is accomplished by sending a "brick" command [13] to the end-point agent, removing the device from the device end-point registry, suspending or revoking PKI certificates (PIV and/or derived credentials) of users, disabling Wi-Fi, and disabling email and other functionality.

Disenfranchised devices are wiped [14] selectively or in full (back to the factory default settings). If ownership of the device is in question, the enterprise end-point manager exercises all of the above. At the discretion of the administrator, a termination notification is sent to the individual(s) of record in the device end-point registry by the method(s) indicated in the device end-point registry. All actions are logged, and termination or disenfranchisement triggers alerts.

A user activates the device, which has PIV readers installed, by using the PIV and passcode. For devices without PIV readers installed, a user authenticates himself to the device, thus binding derived credentials to the user. The device end-point agent provides the user binding to the enterprise. Devices without PIV readers are provisioned with one or more derived credentials (for one or more assigned device users). Authentication and binding of mobile devices typically requires a two-factor authentication since there is no separate hardware device for storage of private keys and the devices are generally physically accessible to non-vetted personnel. The second factor configured for the device is typically biometric (out-of-band is associated with the mobile device) with the biometric determined by device capabilities (face recognition, voice recognition, fingerprint, etc.). The call for second-factor authentication comes from the Security Token Service (STS) upon recognizing the use of a derived credential. Claims are sent to the STS only if the user is coupled to a registered device and the device does not compromise the rules established by the data owner (such as geo-location). Any device that does not have a user/end-point association in the end-point/user dynamic binding store trips an error return (and associated logs and alerts) from the provide claims web service.

Certain devices are configured with multiple derived credentials. This allows multiple users to use a single

ISBN: 978-988-14047-9-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

device. The device end-point agent provides a binding to the user currently logged into the device. Additionally, certain devices provide for device authentication without a certificate. When this occurs, the only option for the user is to proceed to the STS/CA (a special STS with certificate issuing authority {CA}) for the issuance of a temporary certificate through multi-factor authentication. This certificate's private key is installed in the temporary memory of the SKSU for this user on this device. The private key should be encrypted using the public key of the SKSU to ensure only the SKSU can use this software-based private key. The temporary certificate has a short life (currently 90 minutes). This process is described in detail in [15].

IV. Device End-Point Agent

The device end-point agent is software on enterprise-approved devices that interacts with central services. It is a functional element that applies requests from the Enterprise End-point service to the local device and retrieves SKSU data from the local device for the Enterprise End-point service.

The agent is essentially a local mediation service for the SKSU and the central services. It queries the SKSU for the current state of the system, and it communicates with the central services to relay these SKSU reports and other verifiable data. When patches or updates are pushed, the agent applies them locally. Pushed packages come from the end-point manager or, when approved by the enterprise, the application store.

A. Monitoring and Reporting

The agent monitors the status of the device. It periodically queries the SKSU for an attestation report. If such a report is not available or produced in error, the agent alerts the central services. Further action is instructed by the central services, such as disabling certain device functions, removing applications, or completely wiping the device's sensitive data and keys.

Under normal operations, the agent monitors connections and uses these connections to maintain a periodic heartbeat communication with the Enterprise End-point Service. The agent also contacts the service upon initial connectivity and sends an attestation report with the device's status. The device status is "invalid," "current," "current awaiting update," or "not current updates needed." The agent then responds to any requests from the central services for further information or action.

B. Data Validation and Purging

The agent can validate SKSU signatures and data structures, but it cannot be trusted to perform a full validation of local information, because the agent itself can potentially be compromised. For this reason, the agent sends SKSU attestation reports to the central services for further validation against known good values. The central services then directs the agent whether to continue as normal or take corrective actions, such as purging data, keys, and applications from the device.

It is important to note that if a device is stolen and compromised, the agent functionality is compromised as well because it is just another application on the device. Although this should not pose a serious security threat, because the SKSU and its private key(s) should still be secure, it means that a request to wipe the device can never be confirmed. The goal is not complete remote control over the device, but instead to enforce basic compliance rules before allowing users on enterprise-registered devices to connect to enterprise services. The ability of the Enterprise end-point service to revoke access to devices without valid attestation reports mitigates the device itself from becoming a new point of vulnerability.

C. Fulfilling Requests for Data

In addition to standard SKSU attestation, the agent can be queried for other local data available from the device itself or other local services. For example, global positioning system (GPS) location information could be requested or service provider information could be requested. The agent simply relays the information provided and repackages it for consumption by the central services. The agent, as a potentially compromised part of the device, cannot be trusted to relay correct information. Additional security measures, such as digital signatures, are used from the original data providers to guarantee integrity. Disabling signature or other integrity or security functions is considered nefarious behavior and subject to end-point disabling.

The agent for fixed assets (e.g., desktops) within the enterprise reports information about the location of the device. This can be compared against the registered location. This typically includes an address or room number that is static and configured into the machine. Because such devices are within the control of the enterprise, no dynamic location data is needed. An individual can verify the location if needed.

The agent for mobile assets provides location based on best available information (e.g., Wi-Fi access point name, mobile tower identifier, GPS coordinates, altimeter, etc.). Because such devices move frequently and connect from outside the network, dynamic information about location is important for access control or other decisions. The local device is not trusted to provide this information if possible, because it could be compromised. External sources, such as Wi-Fi connection information, GPS data, or wireless tower connection information is potentially valuable location data, but often does not provide security guarantees such as trusted signatures. Due to mobility, the availability of certain types of information is uncertain, so the best effort is made given the current environment and available services. The end-point manager ascertains the veracity of the location measure using logic provided by the enterprise and may place a value of "unknown" in the dynamic file.

V. Conclusions

Management of end-point devices is required for both security and efficiency. In a high-assurance environment, maintaining tight control of both devices and users is

ISBN: 978-988-14047-9-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

mandatory. The formulation is new and being applied to devices within the enterprise. This work is part of a body of work for high-assurance enterprise computing using web services. Basic elements of this work are described in [16]. Advanced techniques are described in [11, and 17-23].

REFERENCES

[1] IBM Corporation, web reference, "Mobile Device Management (MDM)," https://www.ibm.com/security/mobile/maas360/mobile-device-management, last accessed on 18 November, 2017.

[2] AT&T Business, web reference, "CYBERSECURITY SOLUTIONS- Mobile Security," https://www.business.att.com/solutions/Family/cybersecurity/mobile-security/, last accessed on 18 November, 2017.

[3] PC Magazine, web reference, The Best Mobile Device Management (MDM) Solutions of 2017, https://www.pcmag.com/article/342695/the-best-mobile-device-management-mdm-software, last accessed on 18 November, 2017.

[4] MindWireless – Strategic Telecom Management, web reference, "Enterprise Mobility Management," https://mindwireless.com/services/enterprise-mobility-management, last accessed on 18 November, 2017.

[5] Frank Konieczny, Eric Trias and Nevin Taylor, "SEADE: Countering the Futility of Network Security," Air and Space Power Journal, Sep–Oct 2015, Vol 29, No. 5, p. 4.

[6] Technical Profiles for the Consolidated Enterprise IT Baseline, release 4.0. Not available to all, https:// intelshare.intelink.gov/sites/ afceit/TB

[7] TPM Main Specification Version 1.2, Revision 116, 1 March 201, TCG Published, available at: https://trustedcomputinggroup.org/wp-content/uploads/ TPM-Main-Part-1-Design-Principles_v1.2_rev116_01032011.pdf

[8] Ferraiolo, H. , et al, NIST Special Publication 800-157, "Guidelines for Derived Personal Identity Verification (PIV) Credentials, December 2014, http://dx.doi.org/10.6028/NIST.SP.800-157

[9] PC Magazine, Encyclopedia, web reference, "Definition of heartbeat," last accessed on 15 November 2017, https://www.pcmag.com/encyclopedia/term/44190/heartbeat

[10] James W. Hong, et al., Enterprise Network Traffic Monitoring, Analysis, and Reporting Using Web Technology, Journal of Network and Systems Management, Vol. 9, No. 1, 2001

[11] W. Stallings, SNMP, SNMPv2, SNMPv3, and RMON 1 and 2, Third Edition, Addison-Wesley, 1999.

[12] ITU-T, Information Technology, Common Management Information Protocol (CMIP)–Part 1: Specification, Recommendation X.711, 1991.

[13] Techopedia home dictionary tags, web reference, "Bricking-definition and explanation," last accessed on 11/15/2017, https://www.techopedia.com/definition/ 24221/bricking

[14] TechTarget Search Mobile Computing, web reference, "Remote wipe," last accessed on 11/15/2017, http://searchmobilecomputing.techtarget.com/definition/remote-wipe

[15] William R. Simpson, and Kevin E. Foltz, Lecture Notes in Engineering and Computer Science, "Assured Identity for Enterprise Level Security," Proceedings of the World Congress on Engineering, July 2017, Imperial College, London, pp. 440–445,

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

[17] 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.

[18] 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.

[19] 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.

[20] 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.

[21] 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.

[22] 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.

[23] 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.

ISBN: 978-988-14047-9-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)