Managed Certificate Whitelisting – A Basis for Internet of Things Security in Industrial Automation Applications

Rainer Falk and Steffen Fries
Siemens AG
Corporate Technology
Munich, Germany
Email: {rainer.falk|steffen.fries}@siemens.com

***Abstract*—Device authentication is a basic security feature for automation systems and for the future Internet of Things. The design, setup and operation of a practically usable security infrastructure for the management of required device credentials – as cryptographic device keys and device certificates – is a huge challenge. Also, access permissions defining authorized communication peers have to be configured on devices.**

The set-up and operation of a public key infrastructure PKI with registration authority (RA) and certification authority (CA), as well as the management of device permissions has shown to be burdensome for industrial application domains.

A recent approach is based on certificate whitelisting. It is currently standardized for field device communication within energy automation systems by IEC 62351 in alignment with ITU-T X.509. This new approach changes the way how digital certificates are used and managed significantly.

After describing the new approach of managed certificate whitelisting and giving a summary of ongoing standardization activities, an example for the application in a real-world application domain is described. Needs for further technical work are derived, and solution options are presented.

Keywords*—*Digital certificate, certificate whitelisting, credential management, PKI, device authentication, Internet of Things.

I. Introduction

Industrial automation systems, e. g., for energy automation, railway automation or process automation, use open communication protocols as Ethernet, wireless local area network (WLAN) IEEE 802.11 [1], transmission control protocol (TCP), user datagram protocol (UDP), and hypertext transfer protocol (HTTP) [2]. The communication can be protected using standard security protocols like IEEE 802.1X/MACsec [3], Internet key exchange (IKE) [4] with Internet protocol security (IPsec) [5], secure shell (ssh) [6], secure sockets layer (SSL) [7], and transport layer security (TLS) [8]. Often, asymmetric cryptographic keys and corresponding device certificates are used. Symmetric keys would not not scale well for the huge number of involved devices.

In a common realization of a public key infrastructure PKI, digital certificates are issued by a trusted certification authority (CA). This allows to authenticate devices. Additionally, access permissions are defined for authorized communication peers. While this technology could be the basis for a global, uniform secure communication, in reality, the deployment and adoption of PKIs is often limited to HTTP server authentication. A reason for that is the significant effort required to set-up, maintain, and use a PKI.

The problem addressed in this paper is the practical management of device certificates for field-level automation devices. A certificate infrastructure is required that is suitable for an operational automation environment. Main considerations are the demand for extremely high system availability, requiring that the automation system can continue to operate in an autonomous island mode, and the fact that many automation systems are set-up as separate network segments that have no or only limited connectivity with general office networks or even the public Internet. Moreover, the fact that these systems are typically engineered, e.g., that the communication relations are known up front, can be leveraged for certificate and access management.

A self-contained certificate management tool (command line tool, or with GUI) can be well suited for a small number of devices, but it does not scale well to scenarios with a larger number of devices. A full-blown PKI infrastructure could be efficient for an extremely huge number of devices, but these go beyond the scale of a common single automation systems.

The problem can be summarized that a solution is needed that can be set-up and operated autonomously within a certain automation environment without relying on a globally accepted certification authority, and that scales well for “mid-size” automation environments, for which a self-contained certificate tool is too small, and a full PKI solution would be too complex and costly. It may be also advantageous to avoid the need for deploying a separate identity and access management infrastructure.

The remainder of this paper if structured as follows: After summarizing background work in Section II, Section III describes certificate whitelists as a new paradigm for using digital certificates. The management of certificate whitelists is described generically in Section IV, and a specific adaption into energy automation systems is outlined in Section V. An outlook to possible future extensions is given in Section VI.

II. Background and Previous Work

Secure communication protocols, digital certificates, and public key infrastructure PKI [9], [10] have been dealt with intensively for years. An introduction is given in common text books on IT security [11]. The remainder of this section summarizes shortly major aspects that are relevant to managed certificate whitelists.

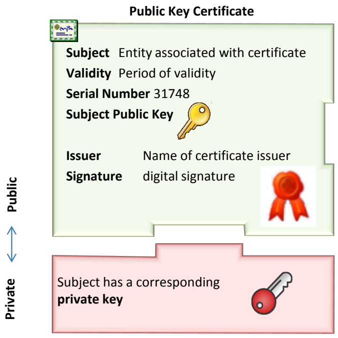
Public Key Certificate
Subject Entity associated with certificate
Validity Period of validity
Serial Number 31748
Subject Public Key
Issuer Name of certificate issuer
Signature digital signature
Subject has a corresponding private key
Public
Private

Fig. 1. Digital Certificate (X.509)

A. Device Communication Security Technologies

Digital device certificates are the basis for device communication security as used in industrial automation systems, and in the future Internet of Things (IoT). Major communication security protocols are available for the different layers of the communication protocol stack that support digital device certificates for authentication:

- Link layer: The standard 802.1X [3] provides Network Access Control to restrict access to a network only for authenticated devices. It is also possible to encrypt the communication link using the MACsec of 802.1X.
- Network layer: The communication can be protected with IPsec [5] on the network layer. The required security associations can be established by the IKE [4] protocol.
- Transport layer: With TLS [8], the successor of the SSL protocol [7], communication can be protected on the transport layer.
- Application layer: SSH or WS-Sec are available to protect application layer protocols as HTTP, SOA (REST, SOAP), CoAP, XMPP, or MQTT.

B. Digital Certificates

The main purpose of a digital certificate is to reliably assign information about the subject, i. e., the owner, of a public key. The owner may be identified by its name or email address in case of a person, or by its network name (DNS name) or IP address of a server. Additional information encodes usage information about the public key respectively the digital certificate, as validity period, and allowed key usages as user authentication or email encryption. For device certificates, it is possible to encode the device manufacturer, the device model, and the serial number within a device certificate.

The most commonly used certificate format is ISO X.509 [9]. Figure 1 shows the format and some examplary fields. The main purpose of a digital certificate is to bind a public key (`Subject Public Key Info`) of an entity to the name of the entity (`Subject`). Additional information as the validity period, the issuer, and usage restrictions can be included as well.

When a digital certificate of a subject is validated by a communication peer, it is verified that the certificate has a valid digital signature of a trusted certification authority. It is furthermore verified that the entries of the certificate match the intended usage. It may also be verified whether the certificate has been revoked. A revocation check may verify whether a given certificate is included in a certificate revocation list (CRL), or an online revocation status check may be performed using the open certificate status protocol (OCSP) [12]. In either case, at least partial online access to a PKI entity that is issuing certificates and providing revocation information is needed at least from one component in an automation network or cell. This component may further distribute the information within the automation cell.

C. Certificate Root Key

A digital certificate has to be validated before it is accepted. This includes a check whether the digital signature protecting the certificate is trusted. The standard approach is to use a set of trusted root certificates for certification authorities CA. A certificate is accepted if its signature chain can be verified back to a trusted root certificate. The root certificate may belong to a globally recognized CA, or to a local CA that is accepted only within an administrative domain, e. g., within a single operator network. If no PKI with CA is available, it is also possible to use self-signed certificates. This means that each certificate is signed with the private key associated with the public key contained in the certificate. Such certificates have to be configured as trusted in the same way as trusted root certificates, i. e., the (self-signed) certificates of trusted peers have to be configured explicitly. This requires to store the trusted peer information (root CA, or self signed certificates) in a secure manner, as this information is crucial for system security.

D. Certificate Whitelisting

The basic concept of certificate whitelists is well-known. The underlying idea is to enumerate explicitly all authorized certificates. A certificate is validated successfully only if it is contained in the certificate whitelist. The whitelist may contain the certificates directly, or reference the certificates by their serial number and issuer, by the certificate fingerprint, or by the public key. The latter avoids issuing a new whitelist, when a certificate is updated.

Such a certificate whitelist can be considered and used also as an access control list that contains the certificates of all authorized subjects. Without using specific certificate extensions, the different operations cannot be distinguished, however. The configuration of the set of trusted root certificates is also a form of certificate whitelists. It is known to check whether the certificate of a communication peer is included in a certificate whitelist [13]. Also, the Microsoft Digital Rights Management License Protocol is using a certificate whitelists [14].

As these certificate whitelists have been used as a proprietary means for configuring a list of trusted certificates, or to be more precise a *set* of trusted certificates, the approach has been rather limited as general means for certificate management.

III. Certificate Management and Validation Using Certificate Whitelists

The set-up and operation of a public key infrastructure has shown to require significant effort and costs. This has been a limiting factor for the practical usage of public key cryptography. Ongoing standardization activities define the technological basis for simpler usage of public key cryptography for industrial automation environments and the future Internet of Things.

While a certificate whitelist has been used so far as proprietary means for configuring some digital certificates as trusted, a certificate whitelists format is currently standardized for the smart energy grid environment. It has been acknowledged that the application of certificate whitelists in restricted environments supports the long term administration of security parameters. Hence, standardizing the format is the next consequent step to ensure interoperability of different vendor products.

A certificate whitelist is a data structure containing respectively referencing a set of trusted digital certificates. A certificate can be referenced by its serial number and issuer, or by a fingerprint of the certificate (hash value). The certificate whitelist is signed using a whitelist root key of trust (WROT).

A certificate is validated successfully if it is contained in a corresponding certificate whitelist. Further checks on the contents of the certificate as the name of the subject, the certificate extensions, and the certificate signature are performed in the usual way.

Certificate whitelists can be used with certificates issued by a CA, or with self-signed certificates. A common technological basis is provided for smaller environments using self-signed certificates as well as environments using a PKI for issuing certificates. So, a smooth migration from self-signed certificates to a local PKI and even towards global PKI is provided.

A certificate can be revoked easily by not including it anymore in the certificate whitelists. However, it is also possible to check the certification revocation status using certificate revocation lists [9] or using the online certificate status protocol OCSP [12].

1) Standardization Activities: Currently ongoing standardization activities performed by ISO/IEC 62351 [15] in alignment with ITU-T X.509 [9] define the usage of certificate whitelists for energy automation systems. Currently, a format is defined for a certificate whitelist. Figure 2 shows a recent proposal for a certificate whitelist. It is based on the format of a certificate revocation list CRL, but its assigned type (`CertificateWhiteList`) distinguishes it from a CRL. Also, the intended scope of a certificate whitelist is defined by a specific attribute `scope`. It allows a client to verify whether a certain certificate whitelist has in fact been intended for a specific purpose. For example, the IP addresses or DNS names of devices for which the whitelist is intended to be used can be included.

The target scope of a certificate whitelist can be explicitly encoded in a certificate whitelist. Therefore, a certificate whitelist cannot be used unintentionally for a different purpose as the intended purpose at time of compilation. Certificate whitelists can be compiled once during as part of engineering. Alternatively, end devices can pull a certificate whitelist from a whitelist certificate server in defined time intervals. The CWL can also be pushed to the field devices.

```
CertificateWhiteList  ::=  SEQUENCE  {
     tbsCertWhiteList     TBSCertWhiteList,
     signatureAlgorithm   AlgorithmIdentifier,
     signatureValue       BIT STRING
}
TBSCertWhiteList  ::=  SEQUENCE  {
   version      Version OPTIONAL,
                -- if present must be v1
   signature    AlgorithmIdentifier,
   issuer       Name,
   thisUpdate      Time,
   nextUpdate      Time OPTIONAL,
   scopedList      SEQUENCE OF SEQUENCE {
      scope        ScopeConstraints,
                   -- geographic,organizational
      authorizedCertificates    SEQUENCE OF SEQUENCE  {
         fingerprint     AlgorithmIdentifier, -- for FP creation
         certIdentifier::== CHOICE {
         serialCert      [0] CertificateSerialNumber,
         fingerprintCert [1] OCTET STRING -- FP of certificate
         fingerprintPK   [2] OCTET STRING -- FP of public key
   }
   certificateIssuer     Name OPTIONAL,
   cwlEntryRestriction   [0] EXPLICIT Extension OPTIONAL
                         -- further restrictions of cert. usage
   }
   }
   cwlExtensions [0] EXPLICIT Extensions OPTIONAL
                  {- for future use
}
```

Fig. 2. Certificate Whitelist Format [15]

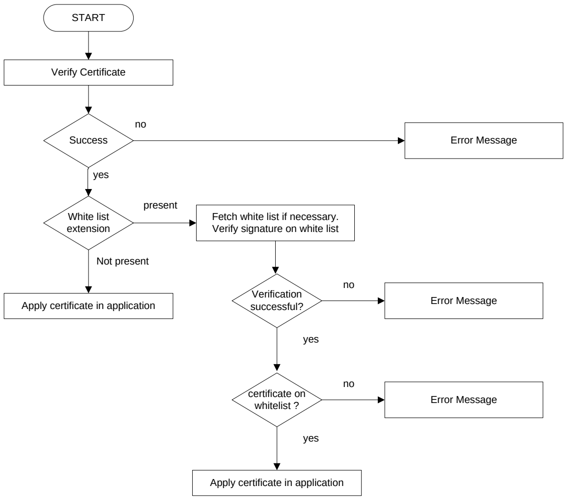
START
Verify Certificate
Success
no
Error Message
yes
White list extension
present
Fetch white list if necessary. Verify signature on white list
Not present
Apply certificate in application
Verification successful?
no
Error Message
yes
certificate on whitelist ?
no
Error Message
yes
Apply certificate in application

Fig. 3. Validation of a Certificate with Certificate

A digital certificate may be intended to be used only within a certificate whitelisting environment. To ensure that a certificate is in fact validated successfully only together with a corresponding whitelist, it is possible to include a corresponding extension in the certificate. The extension marks it explicitly to be accepted only if it is included in a certificate whitelist. A corresponding certificate extension is currently defined by ISO/IEC 62351 [15].

The validation of a certificate depends on whether it contains a certificate whitelist extension. Figure 3 shows the relevant checks. If a certificate includes the whitelisting extension, it is required that the corresponding whitelist is available and that the certificate is in fact included in the whitelist.

IV. Managed Certificate Whitelists

The introduction of certificate whitelisting implies the need for a management system for certificate whitelists. Managed certificate whitelists are a new approach for using public key cryptography in a practical, efficient and effective way. It is particularly suited for systems with well-known set of devices and their communication relationships, as it is common for networked automation systems. As the management of whitelists can be fully automated, it scales well to larger number of devices, although due to the increasing size of whitelists the targeted application environment is characterized by a number of devices within a range up to some 100 to some 1000 devices. It integrates well within existing industrial workflows for installing or exchanging devices, as device configuration databases are kept up-to-date within automation systems. So, the information that is required to generate updated certificate whitelists is already available. Once certificate whitelists have been generated and installed on the target devices, the target devices can operate autonomously even if the security infrastructure is not available. This is an important property for automation environments with high availability requirements to ensure that the automation system can continue to operate even if backend systems are temporarily unavailable.

A. Whitelist Generation and Distribution

The basic concept for automatic whitelist management is rather straightforward. Using information which is available in common automation systems about the devices and their communication relationships within a networked automation system, several purpose-specific – and also device-specific if needed – certificate whitelists are generated automatically. The whitelists are distributed to the target devices using remote configuration protocols. For example, secure copy scp [6], HTTPS [16], or OPC-UA [17] can be used to distribute configuration files securely to the target devices.

Figure 4 shows the main components involved in the automatic management of certificate whitelists. A central device management component accesses a device database including all registered devices of a networked automation system and their associated device certificates. Using automation system configuration data, the communication relationships are determined. Based on this information, certificate whitelists can be compiled for the different communication purposes as automation control communication, supervisory control communication, remote service access and diagnostic access. Depending on policy, device-specific certificate whitelists can be compiled, or certificate whitelists for defined purposes and target device classes. The certificate whitelists are created and provided to a device management system that configures the relevant certificate whitelists on the target devices. As important difference to a certification revocation list CRL, a certificate whitelist will usually be provided and be signed by the operator, not by the certification authority (CA). This has the advantage that an automation system operator can use managed certificate whitelists easily with certificates issued by different CAs, and even with self-signed certificates.

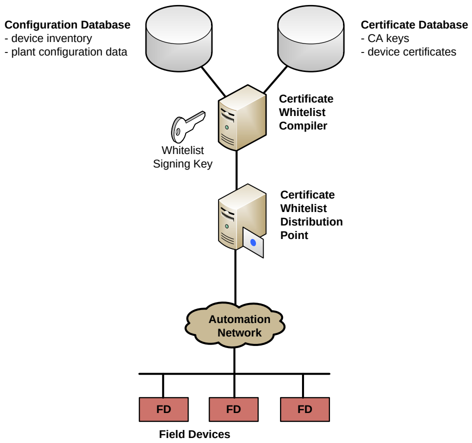

Fig. 4. Certificate Whitelist Management System

For networked automation systems with a typical size of some 100 to some 1000 devices, such a certificate management system based on whitelisting provides several advantages for the application in real-world industrial usage scenarios: A local PKI or even self-signed certificates can be used, so that a deployment with a very limited security infrastructure is possible. For the operation of the automation system, no continous reachability or availability of the whitelisting security infrastructure is required. So, the availability of the automation system availability does not depend on the availability of the security infrastructure. A commonly availably device management infrastructure can be extended easily for automatically creating and distributing certificate whitelists. It is possible to use a certificate whitelist only for authentication. Authorization checks would then be performed in addition, e. g., by checking an access control list. However, a certificate whitelist can be used directly as access control list as well. Different certificate whitelists would be configured for the different types of access (e. g., control communication, service access, diagnosis). The current proposal for a CWL structure considers this by supporting the encoding of a list of lists. Moreover, within the CWL, further certificate usage restrictions may be encoded. One example is the definition of dedicated applications or communication protocols which are allowed to utilize a dedicated certificate. Using this approach, the communication peer could refuse to accept a certificate included on the CWL if it is not associated within the CWL with the currently used communication protocol.

This has the advantage that no separate identity and access management infrastructure is needed, and that access control decisions can be performed by a field device when the backend systems are not available. These properties make certificate whitelisting a very interesting approach for managing digital certificates in typical industrial automation systems.

B. Example Usage Scenarios

Typical workflows in industrial automation systems are the initial installation, the replacement, and removal of devices. As device configuration databases are already maintained as part of these workflows, the information for updating certificate whitelists is available without any extra effort required from the service personnel. As changes in the configuration are detected by the certificate whitelisting system, the generation of updated certificate whitelists is started and the deployment to affected target devices is triggered.

V. Application Within Energy Automation Systems

The general approach of using managed certificate whitelists as described in the previous section can be applied for energy automation systems (smart grid). Figure 5 shows a substation automation system. A substation typically transforms voltage levels, and includes power monitoring and protection functions. Figure 5 shows separate network zones of the substation communication network. The field devices that perform the actual field level functionality of monitoring and acting on the electric power are called intelligent energy devices (IED). They are monitored and controlled by a substation controller, realizing a realtime automation system. Energy automation protocols are defined by the standard IEC61850 [18] which defined the Generic Object Oriented Substation Events (GOOSE) protocol. Additional network zones are available for local and remote service access, for integrating intelligent field devices with serial interfaces, and for support functions (file server, historian server for logging, remote access server, terminal server). A substation is connected to the utility communication network providing backend services like supervisory control and data acquisition (SCADA). Firewalls are used to control the traffic flow between zones.

A hierarchical creation and distribution of certificate whitelists to a substation may be realized in the following way: A utility operator creates a substation-specific certificate whitelist (substation cert whitelist) based on the engineering information for this substation and distributes it to the substation controller. The specific substation is encoded in the CWL by the scope restriction. Using engineering information that is available at the substation controller, the substation controller creates device-specific certificate whitelists for the field devices, i. e., intelligent energy devices (IED), of the substation. The device-specific certificate whitelists are configured by the substation controller on the differend IEDs.

An alternative approach would be to compile a CWL for a substation, and to distribute this CWL to all components in the substation via the substation controller. Through the engineering information, each IED will only communicate with other IEDs by means of the engineering data and the CWL. This means that the access control decision is made by an IED by checking both the CWL and the engineering information. This saves the additional effort for creating device specific CWLs, but has the disadvantage that each IED needs to search a larger CWL, and has to check two pieces of configuration information separately. It is a validation perfomance decision which approach is more appropriate in a target environment. The generic definition of CWLs allows for both approaches.

A further usage scenario for certificate whitelisting within energy automation systems would be integration of decentralized energy resources. Here, a smart grid operator could realize a (managed) certificate pinning by using certificate whitelists. A smart grid operator would define which certificates are acceptable by including these certificates in a whitelist. Thereby, the smart grid operator would use certificate whitelists to restrict the set of certificates issued by a larger PKI. The possibility to misuse broken certificates or CAs is reduced as the set of accepted certificates is limited.

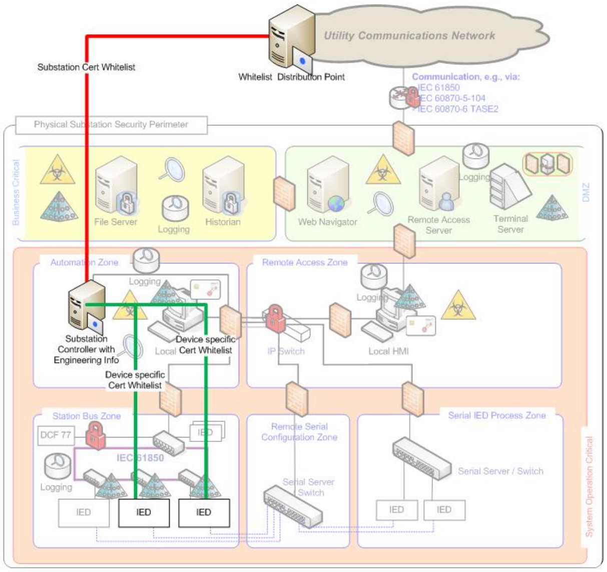

Fig. 5. Managed Certificate Whitelisting in Energy Automation Substations

VI. Conclusion and Outlook

Explicitly designating trusted certificates in certificate whitelists has been recently put forward within standardization for industrial energy automation communication [15]. It promises to provide a cost-efficient, easily deployable, and operable approach for digital device certificates even if self-signed certificates are used. It is intended for mid-sized industrial automation domains, while providing a migration path to more flexible PKI and access management structures. It allows in particular to avoid the usage of simple manually configured pre-shared secrets, that would be difficult to migrate to more complex and managed security infrastructures that are expected to be advantageous for large scale deployments.

The usage of certificate whitelisting can be supported with automatic whitelist generation and distribution. A format for certificate whitelists is currently being standardized to provide an interoperable format. Specific extensions can mark a certificate explicitly for being used only in combination with a certificate whitelist. Several additional extensions may be introduced. It may be possible to indicate usage restrictions within a certificate whitelist associated with a certain certificate entry. This could be used to limit the authorized usage of a certificate on a certificate-by-certificate basis. Certificate whitelists may be encoded efficiently by including matching criteria of included certificates. Alternatively to the explicit enumeration of certificates, a filter can be included in a certificate whitelist that defines matching criteria of included certificates, i. e., that defines required properties of certificate fields. A Bloom filter [19] may be used, combined with a check on false match. Bloom filters are a probabilistic data structure for membership queries which allow for an efficient encoding, but for which a wrong positive match may occur. As the set of all issued certificates is known in typical usage scerarios, a checking for a false match is easily possible. Also, certificates can be designated within a whitelist. Also, a PKI gateway can be deployed for secure interworking with external network domains using a standard public key infrastructures.

Also, the logical combination of multiple certificate whitelists is possible in general. A combination of certificate whitelists may be advantageous for instance in an inter-substation communication scenario. Here, a first certificate whitelist may be provided for the substation internal communication, and a second one for inter-substation communication. The final certificate whitelist for each purpose may be defined by a logical combination of whitelists to ease the certificate whitelist administration and the handling for the field device. This might be done by logical OR, AND, or XOR combinations of the certificate whitelists. This logical combination can be realized in different ways: The field devices themselves can check against multiple certificate whitelists. A logical expression is configured that defines the logical combination of the certificate whitelists to be applied. As the defined certificate whitelist structure shown in Fig. 2 allows the encapsulation of multiple certificate whitelists within a single data structure, an enhancement of this data structure could indicate the logical combination of the whitelist entries using the extension option. A further alternative would be the preparation of device specific certificate whitelists by a centralized infrastructure component that determines the result of the logical combination of different certificate whitelists before distributing the actual certificate whitelist to the end points. This puts more effort on the centralized component, but keeps the effort low for the field device. The assumption here is that the certificate whitelist for a single endpoint is rather short compared to substation wide certificate whitelists containing all allowed (engineered) combinations of communication associations. The structure defined in Fig.2 also allows to use different matching criteria for the certificate. While the serial number and issuer or the fingerprint are straight forward, the utilizatin of the public key fingerprint provides another degree of freedom. This approach allows even for updating certificates (assumed the public key stays the same) without changing the CWL. This decouples the certificate life cycle management from the access security policy management of certificates in automation environments.

REFERENCES

[1] IEEE 802.11, “IEEE Standard for Information Technology– Telecommunications and Information Exchange Between Systems, Local and Metropolitan Area Networks–Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.” [Online]. Available: http://standards.ieee.org/about/get/802/802.11.html [accessed: 2014-09-01]

[2] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Barners-Lee, “Hypertext Transfer Protocol – HTTP/1.1,” 1999, Internet Request for Comments RFC2696. [Online]. Available: https://tools.ietf.org/html/rfc2696 [accessed: 2014-09-01]

[3] IEEE 802.1X-2010, “IEEE Standard for Local and metropolitan area networks–Port-Based Network Access Control,” . [Online]. Available: http://standards.ieee.org/findstds/standard/802.1X-2010.html [accessed: 2014-09-01]

[4] C. Kaufmann, P. Hoffman, Y. Nir, and P. Eronen, “Internet Key Exchange Protocol Version 2 (IKEv2),” Sep. 2010, Internet Request for Comments RFC5996. [Online]. Available: https://tools.ietf.org/html/rfc5996 [accessed: 2014-09-01

[5] S. Kent, and K. Seo, “Security Architecture for the Internet Protocol,” Dec. 2005, Internet Request for Comments RFC4301. [Online]. Available: https://tools.ietf.org/html/rfc4301 [accessed: 2014-09-01]

[6] T. Ylonen, and C. Lonvick, “The Secure Shell (SSH) Protocol Architecture,” Jan. 2006, Internet Request for Comments RFC4251. [Online]. Available: https://tools.ietf.org/html/rfc4251 [accessed: 2014-09-01]

[7] Netscape, “SSL 3.0 specification,” Nov. 1996. [Online]. Available: http://web.archive.org/web/20080208141212/ http://wp.netscape.com/eng/ssl3/ [accessed: 2014-09-01]

[8] T. Dierks, and E. Rescorla, “The Transport Layer Security (TLS) Protocol Version 1.2,” Aug. 2008, Internet Request for Comments RFC5246. [Online]. Available: https://tools.ietf.org/html/rfc5246 [accessed: 2014-09-01]

[9] ITU-T X.509, “X.509 Information technology – Open Systems Interconnection -– The Directory: Public-key and attribute certificate frameworks,” 2012, version 3 corrigendum 3. [Online]. Available: http://www.itu.int/rec/T-REC-X.509-201210-S!Cor3/en [accessed: 2014-09-01]

[10] D. Cooper, S. Santesson, S. Farrel, S. Boeyen, R. Housley, and W. Polk, “Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile,” May 2008, Internet Request for Comments RFC5280. [Online]. Available: https://tools.ietf.org/html/rfc5280 [accessed: 2014-09-01

[11] J. Buchmann, E. Karatsiolis, and A. Wiesmaier, “Introduction to Public Key Infrastructures,” 2013.

[12] S. Santesson, M. Myers, R. Ankney, A. Malpani, S. Galperin, and C. Adams, “X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP,” Jan. 2013, Internet Request for Comments RFC6960. [Online]. Available: https://tools.ietf.org/html/rfc6960 [accessed: 2014-09-01]

[13] eTutorials.org, “C/C++ Secure Programming – Chapter 10.9 Using a Whitelist to Verify Certificates,” 2014, eTutorials.org. [Online]. Available: http://etutorials.org/Programming/secure+programming/ [accessed: 2014-09-01]

[14] Microsoft, “Digital Rights Management License Protocol – Retrieving Revocation Data from the Enrollment Server,” 2014. [Online]. Available: http://msdn.microsoft.com/en-us/library/dd644914.aspx [accessed: 2014-09-01]

[15] ISO/IEC 62351, “Power systems management and associated information exchange Data and communication security,” 2014, IEC TC57. [Online]. Available: http://tc57.iec.ch/index-tc57.html [accessed: 2014-09-01]

[16] E. Rescorla, “HTTP Over TLS,” 2000, Internet Request for Comments RFC2818. [Online]. Available: https://tools.ietf.org/html/rfc2818 [accessed: 2014-09-01]

[17] OPC Foundation, “OPC Unified Architecture Specification Part 1: Overview and Concepts, Release 1.02,” Jul. 2012. [Online]. Available: http://www.opcfoundation.org/ua/ [accessed: 2014-09-01]

[18] ISO/IEC 61850, “IED Communications and Associated Data Models in Power Systems,” 2014, IEC TC57. [Online]. Available: http://tc57.iec.ch/index-tc57.html [accessed: 2014-09-01]

[19] Wikipedia, “Bloom Filter.” [Online]. Available: http://en.wikipedia.org/wiki/Bloom_filter [accessed: 2014-09-01]