2016 NDIA GROUND VEHICLE SYSTEMS ENGINEERING AND TECHNOLOGY SYMPOSIUM
VEHICLE ELECTRONICS AND ARCHITECTURE (VEA) TECHNICAL SESSION
AUGUST 2-4, 2016 – NOVI, MICHIGAN

STRONG AUTHENTICATION IN VICTORY USING PKI

Bob Fedorchak
CACI Supporting US Army CERDEC S&TCD CSIA
Aberdeen Proving Ground, MD

ABSTRACT

This paper focuses on the use of PKI within intra vehicle networks in compliance with the VICTORY specification. It will describe how the use of PKI within vehicle networks can leverage and integrate with the other PKI efforts across the Army to ensure a consistent and interoperable solution. It will also describe some of the challenges with implementing PKI as part of VICTORY and introduce possible solutions to address these challenges.

INTRODUCTION

PKI is one of the critical security technologies that is used to protect networks and data. It provides Authentication, Confidentiality, Data Integrity, and Non-Repudiation, which are key capabilities that are required for cyber operations. Recently, the Office of the Secretary of Defense (SECDEF), the Defense Department Chief Information Officer (DoD CIO), U.S. Cyber Command (USCYBERCOM) and the U.S. Army Cyber Command (ARCYBER) have released memorandums directing the acceleration of PKI implementation in order to improve security on DoD networks. With PKI compliance being tracked across the DoD and lack of PKI compliance impacting the ability for systems to obtain an Authority to Operate (ATO), the need for PKI at the tactical level has become critical.

Systems operating in Army tactical environments are resource constrained and need to be able to operate in a disconnected, intermittent or limited (DIL) environment that is independent of any fixed infrastructure or services. As such, deploying PKI solutions at the tactical level must overcome a number of challenges including limited bandwidth; high latency; intermittent or disconnected networks; limited size, weight, and power (SWaP); and environmental conditions such as wind, dust, dirt, rain, and heat that require ruggedized equipment. Additionally, any PKI solution that is fielded to tactical systems needs to be as transparent to the soldier as possible so as not to impact the performance of the mission.

The Vehicular Integration for C4ISR/EW Interoperability (VICTORY) Standard Specification v1.6.2[1] is a technical specification for the integration of Command, Control, Communications, Computers, Intelligence, Surveillance, Reconnaissance / Electronic Warfare (C4ISR/EW) and other electronics equipment on U.S. Army ground vehicles. Within the VICTORY specification, services, such as the Data Signing and Signed Data Verification Services, either require or recommend the use of PKI as a mechanism for improving security within the vehicle network. While the VICTORY specification assumes the use of a PKI, it does not discuss the many details necessary for a successful implementation of PKI.

Any implementation of PKI as part of the VICTORY standard needs to include both the management and the use of the cryptographic credentials provisioned from the PKI. The PKI management operations include the provisioning or issuance of PKI credentials as well as revocation of these credentials in the case of compromise or overrun. Revocation is expected to be performed as an out-of-band operation (i.e., it is not performed from within VICTORY). After all, if the system is compromised or the vehicle is in enemy hands, any revocation request coming from a VICTORY component could not be trusted anyway. As such, revocation is not discussed in this paper. The VICTORY specification describes which components and services can use PKI and the type of PKI operation (e.g., authentication, data signing etc.) that is performed. One of the operations that is not described under VICTORY is how to validate the PKI credential. This paper focuses on how to address these two gaps: how to issue credentials to VICTORY components and PKI certificate validation operations.

In order to support successful mission operation, the PKI implementation for VICTORY must consider the unique tactical environment in which the vehicle must operate. One

of the key constraints is that the vehicle must be able to operate completely disconnected from any network. If there is network connectivity, it is assumed to be shared by various systems and the connection is expected to be intermittent with limited bandwidth and high latency. The end result is that any PKI-based solution needs to be able to operate independent of any systems, services or components that exist outside of the vehicle.

USING PKI

Within VICTORY there are three key IA services that utilize PKI: the Data Signing Service (DSS), the Signed Data Service (SDS), and the Authentication Service. When a component needs to digitally sign information, it sends the data to be signed along with the PKCS#12[2] file and associated password to the DSS. PKCS#12 defines an archive file format for storing many cryptography objects as a single file. It is commonly used to bundle a private key with its X.509 certificate or to bundle all the members of a chain of trust. PKCS#12. The DSS computes a message digest or hash of the data and uses the password to access the private key in the PKCS#12 file. The private key is used to create the digital signature by encrypting the hash. The DSS sends the digital signature back to the requesting component.

When a component receives or accesses signed data, the data needs to be checked to verify that it has not been modified and that it originates from a trusted source. To do this the component sends the data and the digital signature to the SDS. The SDS computes a hash of the data it received using the same hashing algorithm as the signer. The SDS then uses the public key from the certificate to decrypt the hash that was encrypted by the signer. The SDS compares the decrypted hash to the hash computed from the data that it received. If the hashes match, the data has not been modified since it was signed. Since the encrypted hash was able to be decrypted by the certificate, it proves that the data was signed by the private key associated with the certificate. Since the private key associated with the certificate is only known to the signer, it proves that the data came from the signer.

Validating Certificates

Each time an X.509 PKI certificate is used, the certificate must be validated to determine if it can be trusted. This trust decision has two basic parts: the determination of whether or not the certificate comes from a trusted source and whether or not the certificate is valid. To do this the Relying Party sends a reference to the PKI certificate associated with the digital signature to the Authentication Service. This reference can be the certificate itself, the Subject Key Identifier (SKI) from the certificate, or the Issuer and Serial Number from the certificate. Any one of these three mechanisms uniquely identifies a certificate. Regardless of the mechanism used to uniquely identify the certificate, the certificate will need to be validated as per RFC 5280[3]. This includes a few key operations:

- Building a path
- Verifying the Validity Period
- Verifying the Signature
- Verifying the Extensions
- Checking the Revocation Status.

In order to build a path, the Relying Party must have a set of CAs that are explicitly trusted. This explicitly trusted set of CAs is known as Trusted Authorities or Trust Anchors. In order to prove that the certificate is part of a trusted PKI, the certificate must be issued by a CA that is trusted. The issuing CA must be explicitly trusted or must in turn be issued by a CA that is trusted and so on, until one of the issuers is identified as an explicitly Trusted Authority. In this fashion the certificate can be traced back to a Trusted Authority, which tells the Relying Party that the certificate was issued in accordance with a set of approved issuance procedures. Identifying the chain or path from the certificate to a trusted issuer is known as path building or chaining. Once a path to a Trusted Authority has been established, it is important to verify that each certificate in the path is still valid. If the certificate cannot be traced back to a trusted issuer or if any certificate in the path is no longer valid, the certificate should not be trusted and any operation being performed with that certificate should be terminated. During path building the trusted certificates can be retrieved from the Authentication Service as described in section 17.3.5.1.5 of the VICTORY specification[1].

Every certificate has a start and end date, known as the certificate's validity period. When a certificate is issued by the CA, the CA sets both the start date and the end date for that certificate. If the date and time of the PKI operation using the certificate is outside of the validity period of the certificate (i.e., before the start date or after the end date), the certificate is considered invalid and should not be trusted. Each certificate in the trusted path needs to be checked to determine if the certificate is still within its validity period.

At the time the certificate is issued, the certificate is digitally signed using the CA's private key. By verifying the digital signature on the certificate, the Relying Party confirms that this certificate has not been modified and that it was issued by the CA that signed the certificate. If the data in the

certificate has been modified in any way (e.g., someone tampers with the data in the certificate) or if the signature was created by a private key other than the CA's private key (e.g., someone else tries to "forge" the CA's signature), the signature will no longer verify. Therefore, if the signature does not verify the certificate is considered invalid and should not be trusted. Each certificate in the trusted path needs to be checked to determine if the certificate signature is valid.

When a CA issues a certificate it can include a set of extensions on the certificate itself. These extensions provide additional information to the Relying Party, such as the policy under which the certificate was issued and the usages allowed by the certificate. Each extension is defined by an Object Identifier (OID) and a set of data. If an extension is marked as critical, the Relying Party must check that extension and confirm that the certificate is being used in accordance with the data in the extension. If a Relying Party encounters an extension that is marked as critical, but the Relying Party does not know how to process the extension, the certificate must be rejected as invalid since the Relying Party is unable to process some information that issuer deemed critically important. The most common example of a critical extension is the key usage extension. The key usage extension defines how the public key in the certificate and its associated private key are to be used. For example, assume a certificate includes the *digitalSignature* key usage, but does not include the *keyEncipherment* key usage. This certificate is allowed to be used to sign documents, but it should not be used to establish a TLS connection because TLS performs a *keyEncipherment* and the certificate is not supposed to be used for *keyEncipherment* operations. Since the key usage extension is marked critical on the certificate, a Relying Party trying to establish a TLS connection (e.g., browser or web service) should reject the certificate because it does not include the *keyEncipherment* key usage. Each certificate in the trusted path needs to be checked to determine if the certificate is being used in accordance with the extensions.

One of the most challenging parts of certificate validation is how to determine the current revocation status of a certificate. Determining the current status of a certificate is important because the status of a PKI certificate can change at any time. For example, if an adversary gains access to the private key associated with a certificate, the certificate status must be changed to REVOKED so that the people and systems that are attempting to use the certificate know that it has been compromised. If a component within VICTORY encounters a certificate that has been revoked it should reject the use of the certificate because it cannot be trusted.

All of the services within VICTORY that rely on PKI-based cryptography will need to be able to perform all five of these X.509 certificate validation operations, including the revocation status of each certificate in the path in order to be compliant with the Risk Management Framework (RMF) IA Controls and DoD and Army PKI policy.

The authoritative source for revocation status is the Certificate Revocation List (CRL) created by the CA that issued the certificate. The CAs regularly publish new CRLs with updated certificate status information, and depending on the CA, the size of the CRL can be quite large. For example, the DoD PKI CRLs are published every 18 hours and are over 160MB in total file size. The combination of regular updates, the large data size, and the fact that the CAs that created the CRLs are external to the vehicle, make revocation checking a challenge for VICTORY.

Certificate Validation Service

A Certificate Validation Service (CVS) is needed in order to ensure that the certificate validation process is performed in a consistent and common manner, as well as to provide certificate revocation status information to Relying parties. The CVS can either be implemented as part of the Authentication Service or as a separate service. For this paper the capability is assumed to be a separate service that is invoked by the Authentication Service to obtain updated certificate status information. This is a modular approach that enables any Relying Party component, including the Authentication Service itself, to benefit from the use of the CVS. There are a number of products that could be used for this purpose and further information about viable products can be found in the Army Online Certificate Status Protocol (OCSP) Trade Study [4].

For each CA in the certificate trust store that is part of the VICTORY Authentication Service, the CRL for that CA will be loaded into the CVS during system install time. This ensures that revocation status information is always available for every certificate that is used within the VICTORY Data Bus (VDB), even when there is no external network connectivity. The issue is that any CRLs loaded at install time will quickly become out of date, requiring new CRLs to be provided in order to obtain updated revocation status information. Given the size of the CRLs and the DIL environment in which the vehicle may be operating, it is unlikely that new CRL information can be successfully obtained with enough frequency to adequately support the ground combat vehicle's operating environment.

One possible way to address this is for the CVS to use an RFC 6960[5] compliant OCSP implementation to reach an

OCSP service. OCSP services, such as the DISA Robust Certificate Validation Service (RCVS) or the Army's OCSP services at Brigade and Battalion contain regularly updated CRL information and also provide OCSP responses. By using the OCSP services within the Army Brigades it enables updated information to be obtained without requiring updated CRLs and without requiring reachback to DoD or Army Enterprise services. Using Enterprise services, such as the DISA RCVS, from within Army vehicles can add significant delay (e.g., 18 to 19 seconds as tested at NIE 15.2[6]) and may not contain the necessary revocation information since RCVS only contains DoD PKI and National Security System (NSS) PKI CRLs. If the certificate being checked is not from the DoD PKI or NSS PKI, DISA RCVS will not have the necessary revocation status information. The concern with using the OCSP approach is that it requires connectivity to an OCSP service at the time the certificate is being validated. As such, the CVS would need to be able to reach the OCSP service at any time, which may not be always be possible given tactical network constraints.

To address these concerns with updated revocation status information, the CVS will need to support a combination of technical approaches to obtain updated information in a more robust and reliable manner:

- The ability to locally load the CRLs via removable media or direct network connection will ensure that an out of band load can occur. This is useful for cases such as a quarterly update or regular maintenance activities where new CRLs can be loaded along with any new software patches and configuration files etc. It is also useful for cases where the vehicle does not have connectivity to other networks.

- The ability to support OCSP from an external OCSP Responder will enable the CVS to obtain more recent status information from OCSP services deployed within Army Brigades without needing to load a full set of updated CRLs. This will only be possible if there is a network connection to the external OCSP service.

- The ability to load over-the-air (OTA) enables CRLs to be updated on a more regular basis when the vehicle is forward deployed. As shown during C4ISR Ground Activity Exercise 2015[7] it is technically possible for CRL downloads to be performed over the air using Soldier Radio Waveform (SRW) and Wideband Network Waveform (WNW), even for the larger DoD CRLs. However, since there is a significant impact on the SRW/WNW networks, the use of the OTA CRL update is really only a viable option for very small CRL files. The use of OCSP or locally loaded CRLs is a more viable option for the CVS.

These mechanisms for loading new revocation data combined with the initial CRL loads will help ensure that the VICTORY CVS, and in turn all of the components and services that use PKI, will have the information needed to make trust decisions, even when the vehicle is not connected to external networks.

CREDENTIAL LIFECYCLE MANAGEMENT

In order for X.509 PKI certificates to be used within VICTORY, there needs to be a way to manage the lifecycle of each certificate from the time it is created to the time it is no longer valid. These lifecycle management operations include issuing certificates, revoking certificates and identifying when certificates are going to expire. In order to ensure a common implementation for managing the PKI certificate lifecycle within VICTORY, a PKI Certificate Management Service (PCMS) is being proposed.

The PCMS will provide issuance and monitoring of PKI certificates used by components within the VDB. Certificate revocation is typically performed as an out of band operation when a certificate is lost or compromised and therefore is not part of the PCMS. Any component on the VDB will be able to request a new PKI certificate from the PCMS. Similarly, the PCMS will monitor the PKI certificates being used and will either automatically renew certificates prior to expiration or where automatic renewal is not possible, will log and send alerts that indicate that the certificate needs to be renewed. The PCMS can be implemented as either a separate service or as part of the existing Authentication Service.

Certificate Issuance & Renewal

One of the challenges is how to initially provision and renew the certificates for the components on the VDB. To perform certificate issuance, components on the VDB either need a connection to the issuing CA or an out-of–band method for obtaining and installing the credential. There are a variety of different methods for issuing certificates to components on the VDB:

- ***Direct CA Model***. In this model the CA is external to the VDB and the PCMS generates the key pair locally and creates a Certificate Signing Request (CSR). The CSR is sent to an external CA where the CA uses the CSR to create the certificate. The certificate is returned to the PCMS in a common format, such as PKCS#7[8]. This model requires the VDB to have network connectivity to the

external CA in order to perform the issuance. If network connectivity between the PCMS and the CA is not available, manual intervention will be needed. In the manual case, the CSR must be exported from the PCMS and delivered to the CA via an out of band or "air gap" method. Similarly, the certificate must be returned from the CA to the PCMS using an out of band approach as well. The *Direct CA Model* approach is best used when the VDB has some level of connectivity to external networks and the process can be automated.

- ***Issuance Service Model***. In this model the entire process, including the CA, the key generation, and the formation of the credential, are all performed outside the VDB. A soldier would use the issuance service to generate the keys, communicate with the CA, obtain the certificate and create a PKCS#12 file that contains the private key and associated certificate. The resulting PKCS#12 file would be delivered to the VDB. The delivery may be over the air (where possible) or via an out of band method such as removal media. The *Issuance Service Model* is most useful when the VDB is expected to operate in a DIL environment most of the time, but still needs certificates for both internal and external use.

- ***Self-Generated Model.*** In this model the PCMS generates the keys and the certificates all within the VDB without any dependency on external assets or connectivity. It can use a CA that is part of the PCMS to issue certificates or it can use self-signed certificates. When used within the VDB, the Authentication Service trust store would include each of the certificates generated via this model and would be configured to explicitly trust them. *Self-Generated Model* should only be used for a closed system where the certificates are only used within the VDB. Using this approach for certificates that are used outside of the VDB (e.g., certificates used communication with external components/systems) is not recommended because it requires each of the self-signed certificates to be explicitly trusted on every external device/system with which the VDB components interact.

Where possible the renewal of PKI certificates should be automated by the PCMS. Ideally, the PCMS identifies that a certificate is about to expire and automatically obtains a new certificate from the CA. How well this can be accomplished depends on a number of factors, such as the network connectivity from the VDB to the CA, the CA product being used, which issuance model is implemented, and the issuance policies and procedures required by the PKI under which the CA is operating. Due to the DIL nature of the tactical environment in which the combat vehicles operate, either the *Issuance Service Model* or the *Self-Generated Model* are likely to be the best approach for use within VICTORY.

By centralizing the certificate issuance within the PCMS, it enables all of the VDB components to obtain certificates using a common approach and helps prevent stovepipe solutions from being created to support each component. Since the Authentication Service already needs to have a FIPS 140-2 certified cryptographic implementation in order to support the trust decisions, the PCMS will also be able to leverage the existing FIPS 140-2 certified cryptographic modules provided as part of the Authentication Service.

Certificate Expiration Monitoring with PKITE

Once the *notAfter* date on an X.509 certificate has already passed, the certificate is *expired* and is no longer considered valid. Expired certificates can cause the components and services that rely on the expired PKI certificate to fail. For example, if an existing certificate expires, the component to which the certificate belongs will start to experience problems, such as not being able to establish a secure TLS connection to other components, inability to digitally sign a message, or inability to authenticate. The specific impact depends on what the PKI certificate is being used for, but in any case, the functions (e.g., authentication, session protection, message signature) that rely on PKI certificates will fail once the certificate has expired.

To prevent certificate expiration from causing system problems and negatively impacting the mission, it is necessary to replace the certificates before they expire. Where automatic renewal is available, it should be used. Since automatic renewal is not always possible, the PCMS will need to be able to detect that a certificate is about to expire and alert the soldier or system administrator who can take action to ensure that a new certificate can be installed prior to any mission impact. To do this the PCMS must be able to track all of the X.509 PKI certificates being used within the VDB to determine when they will expire.

The U.S. Army CERDEC S&TCD CSIA Division has developed a tool, known as PKITE, which monitors X.509 PKI certificate expirations. The PKITE tool can be deployed as a separate certificate monitoring service or it can be integrated into the PCMS to provide certificate monitoring to components on the VDB. For the purpose of this paper, this certificate monitoring capability is assumed

to be part of the PCMS. The PKITE solution provides a dashboard that can be accessed via web browser to see the status of all of the certificates being monitored. This enables the soldier to check on the certificate status whenever necessary. However, the intent is for this process to be as automated as possible, so as not to require proactive action on the part of the soldier in order to identify the expiration before it happens. As such, PKITE will be configured to send periodic notifications (e.g., once an hour, once a day once a week etc.) in advance of the expiration date (e.g., 30 days, 60 days in advance). These notifications are formatted as standard syslog events which can be sent to the VICTORY Security Event Log Service (SELS). By integrating with the SELS, this solution provides a consistent way to inform the soldier. Using PKITE on the VDB provides a proactive approach that identifies when a certificate is about to expire and gives the soldier time to react before the VDB components and services fail.

CONCLUSION

PKI is one of the critical security technologies that is used to protect networks and data. With PKI compliance being tracked across the DoD and lack of PKI compliance impacting the ability for systems to obtain an Authority to Operate (ATO), the need for PKI at the tactical level has become critical. While systems operating at the tactical level face some significant challenges in adopting PKI, this paper provides possible solutions to the issuance, monitoring and validation of X.509 PKI certificates in order to address the challenges with implementing PKI as part of the VICTORY standard. It provides options to enable the use of PKI in order to better protect vehicle systems and includes alternatives that take into account the need to prevent this advanced security capability from negatively impacting the vehicle's performance as well as the soldier's mission.

REFERENCES

[1] Vehicular Integration for C4ISR/EW Interoperability (VICTORY) Standard Specifications, Version 1.6.2, March 31, 2015

[2] IETF RFC 7292, PKCS#12: Personal Information Exchange Syntax, version 1.1, July 2014

[3] IETF RFC 5280, Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, May 2008

[4] CERDEC S&TCD CSIA, OCSP Trade Study, December 2015

[5] IETF RFC 6960, X.509 Internet Public Key Infrastructure Online Certificate Status Protocol—OCSP, June 2013

[6] CERDEC S&TCD CSIA, NIE 15.2 Public Key Infrastructure Demonstration Report, July 2, 2015

[7] CERDEC S&TCD CSIA, Tactical Public Key Infrastructure (TPKI) C4ISR Ground Activity Exercise 15 Test Report, December 2015

[8] IETF RFC 2315, PKCS#7: Cryptographic Message Syntax, version 1.5, March 1998