NISTIR 7849

A Methodology for Developing Authentication Assurance Level Taxonomy for Smart Card-based Identity Verification

Ramaswamy Chandramouli

http://dx.doi.org/10.6028/NIST.IR.7849

National Institute of Standards and Technology
U.S. Department of Commerce

NISTIR 7849

A Methodology for Developing Authentication Assurance Level Taxonomy for Smart Card-based Identity Verification

Ramaswamy Chandramouli
Computer Security Division
Information Technology Laboratory

http://dx.doi.org/10.6028/NIST.IR.7849

March 2014

U.S. Department of Commerce
Penny Pritzker, Secretary

National Institute of Standards and Technology
Patrick D. Gallagher, Under Secretary of Commerce for Standards and Technology and Director

National Institute of Standards and Technology Interagency or Internal Report 7849
40 pages (March 2014)

Comments on this publication may be submitted to:

National Institute of Standards and Technology
Attn: Computer Security Division, Information Technology Laboratory
100 Bureau Drive (Mail Stop 8930) Gaithersburg, MD 20899-8930

Reports on Computer Systems Technology

The Information Technology Laboratory (ITL) at the National Institute of Standards and Technology (NIST) promotes the U.S. economy and public welfare by providing technical leadership for the Nation's measurement and standards infrastructure. ITL develops tests, test methods, reference data, proof of concept implementations, and technical analyses to advance the development and productive use of information technology. ITL's responsibilities include the development of management, administrative, technical, and physical standards and guidelines for the cost-effective security and privacy of other than national security-related information in Federal information systems.

Abstract

Smart cards (smart identity tokens) are now being extensively deployed for identity verification for controlling access to Information Technology (IT) resources as well as physical resources. Depending upon the sensitivity of the resources and the risk of wrong identification, different authentication use cases are being deployed. Assignment of authentication strength for each of the use cases is often based on: (a) the total number of three common orthogonal authentication factors – What You Know, What You Have and What You are, and (b) the entropy associated with each factor chosen. The objective of this paper is to analyze the limitation of this approach and present a methodology for assigning authentication strengths based on the strength of pair wise bindings between the five entities involved in smart card based authentications – the card (token), the token secret, the card holder, the card issuer, and the person identifier stored in the card. The rationale for the methodology is based on the following three observations: (a) The form factor of the smart identity token introduces some threats of misuse; (b) the common set of credentials objects provisioned to a smart card embody bindings to address those threats and (c) the strength of an authentication use case should therefore be based on the number and type of binding verifications that are performed in the constituent authentication mechanisms.The use of the methodology for developing an authentication assurance level taxonomy for two real world smart identity token deployments is also illustrated.

Keywords

card issuer; cardholder trait (biometric); person identifier; smart identity token; token secret.

Acknowledgements

The author, Ramaswamy Chandramouli (Mouli), would like to thank his colleagues Hildegard Ferraiolo and Patrick Grother for serving as reviewers for this document. The author also acknowledges Elizabeth Lennon for her technical editing and administrative support.

Audience

This document analyzes the authentication mechanisms used with smart identity tokens based on some fundamental principles in order to derive a metric for assigning appropriate authentication strengths to them. The potential audiences that could benefit from this document are:

- Public and Private Sector communities seeking to deploy smart cards for identity verification (smart identity tokens) for various access control applications;
- Vendor communities seeking to personalize smart identity tokens; and
- Testing communities seeking to evaluate smart identity token deployments for required authentication strengths.

TABLE OF CONTENTS

1. Introduction

With the proliferation of web-based applications and e-commerce transactions, the field of identity verification or authentication of humans has evolved from the concept of using identities tied to a specific entitlement (e.g., a driver's license or passport) to the concept of using generic trusted digital identities that can be relied upon and consumed by multiple types of service providers. Another evolutionary trend is the use of multiple form factors to carry or support these trusted identities. Smart cards and Smart phones are two such form factors.

Smart cards are now being extensively deployed for identity verification for controlling access to Information Technology (IT) resources as well as physical resources [Ham2001, Kum2008, TWIC2008].We refer to those types of cards as Smart Identity Tokens and use the two terms interchangeably throughout this paper. These types of smart cards generally carry: (a) A Person Identifier (PI), (b) A Secret (TS) usually in the form of a cryptographic key [EAG2013], (c) A Credential linking the Secret and the Identifier (CR) and (d) A Credential linking the Identifier with a Personal Trait of the Cardholder (e.g., biometric) (BR). Along with these data, another secret, a PIN (a combination of numbers) is often used for: (a) Activating the card (token) and for (b) Restricting access to certain data objects and operations. In some instances, presentation of a live biometric data (such as a fingerprint) is used to enable the above functions instead of a PIN. In any enterprise deploying smart cards, there may be different types of resources that may have to be protected by restricting access to only those whose identity is verified through a smart card based authentication mechanism. Depending upon the sensitivity of the resource and the risk associated with wrong identification of the entity requesting access to those resources, authentication mechanisms use different combinations of the four data types enumerated above (i.e., PI, TS, CR or BR) along with/without an activation data. One or more of authentication mechanisms in turn constitute an authentication use case and a typical identification verification deployment instance uses multiple authentication use cases to cover access to resources of multiple sensitivity levels.

The choice of an authentication use case (irrespective of whether a smart identity token is used or not) in any deployment instance, therefore, depends upon the overall authentication assurance level provided by the combination of constituent authentication mechanisms.The usage of a token by a claimant during an authentication event results in a value called Authenticator that is generated by the token and is transmitted from the token to the authentication module or the verifier. The basis for designating an authentication strength associated with a token is a fundamental unit called "Authentication Factor". There are three main authentication factors [OMB2003]:

- What the Entity Knows (e.g., Password, PIN, etc)
- What the Entity Has (e.g., possession of a token that generates one-time passwords)
- What the Entity Is (e.g., inherent physiological characteristic such as a fingerprint)

A token that uses one of the above three factors is called a single factor token (e.g., a password that belongs to "What the Entity Knows" factor). A token that uses a combination of two or more of the above factors is called a multi-factor token. A smart card that contains an embedded private cryptographic key (thus using "What the Entity Has" authentication factor) that can be used to generate an authenticator when it is activated by a PIN, (using the "What the Entity Knows" authentication factor) is deemed a two-factor token. An authentication use case may use one or more tokens and hence may involve the use of one or more authentication factors. In general, the authentication strength associated with an authentication use case is determined based on the combination of the following metrics:

- The number of authentication factors used in the authentication use case
- The Entropy associated with each of the authenticator factors used

In this publication, we argue that the logic for assigning authentication strength based on the number of authentication factors in an authentication use case is valid only under certain limiting conditions and that these conditions do not hold in the case of authentication use cases using smart cards as identity tokens. This is the rationale for proposing a new methodology for: (a) Assigning authentication strengths or levels for various authentication use cases involving smart identity tokens and (b) Deriving an authentication assurance taxonomy using the relative strengths of all authentication use cases specified for the deployment.

The limitations of the authentication factor-based approach for determining authentication assurance level and justifications for a new methodology are outlined in Sec. 2. The overall anatomy of smart card-based identity verification is analyzed in Sec. 3. The analysis leads to the identification of bindings established in the initial phases of smart card-based identity verification deployment which then forms the foundational concept for our methodology. The next two sections (Sec. 4 and 5) describe the core steps of our methodology. In Sec. 4, we enumerate the typical set of data objects found in smart cards used in identity verification, the trust bindings each of those objects embodies and the primitive authentication mechanisms that verify those bindings. Section 5 goes on to demonstrate the process of deriving an authentication strength (based on the composition of verified bindings as well as their number and type) for any authentication use case constructed using the primitive authentication mechanisms discussed in Sec. 4. By examining the composition of the "set of verified bindings" in various authentication use cases, it is possible to derive partial orderings among those use cases. These partial orderings, in turn, are used to develop the authentication assurance taxonomy for the total set of authentication use cases specified for a smart identity token deployment. Section 6 provides the conclusions and benefits of our methodology.

In Appendices A and B we demonstrate the use of our methodology to real-world smart identity token deployments. The deployments are: (a) Personal Identity Verification (PIV) program of the US Government and (b) Transporation Worker Identification program (TWIC) of the Department of Homeland Security. More specifically, Appendix A describes the application of our methodology to PIV authentication use cases while Appendix B illustrates our methodology for TWIC authentication use cases. The outcome of the assignment of authentication assurance levels based on our methodology to the complete set of authentication use cases in these two deployments results in an authentication assurance taxonomy for each of them.

2. Limitations of Authentication Factor-Based Approach

In identity verification schemes where trusted identities are provisioned to devices with various form factors (e.g., smart cards, smart phones etc), the authentication factor-based approach for determining authentication strengths (for authentication mechanisms) does not provide the right measure of identity assurance. This is due to the fact that the form factor of the devices introduces some threats of misuse which may not be adequately detected by some authentication mechanisms used in those devices-based identity verification deployments. These threats are briefly described here below. We use the abbreviation convention FF-Tx to designate each threat (FF stands for Form Factor and Tx is the sequence number for the threat)

- FF-T1: STOLEN DEVICE (with unaltered credentials): The person trying to obtain authentication using the device is not the owner of the device/legitimate holder of the credential. This results in "Impersonation" threat.
- FF-T2: CLONED DEVICE (with unaltered credentials): The device containing the credential could be a clone of the device where the original credentials had been provisioned by the legitimate identity provider/credential issuer/authorized device issuer. The threat here is "Unauthorized Proliferation of Credentials and Resulting Misuse."
- FF-T3: FORGED CREDENTIAL: The credential on the device has not originated from an authorized issuer/identity provider. Specifically it does not carry the proof that it was created/assigned by an authorized identity provider and has not been tampered with after issuance.

Thus we see that there is a need for an authentication assurance methodology that takes into account inherent characteristics of the device supporting the trusted identities. Since smart card is the most prevalent device used for provisioning of credentials, we now proceed to analyze the anatomy of smart card-based identity verification in the next section.

3. Anatomy of Smart Card-Based Identity Verification

Smart card-based identity verification is the most widely deployed form of device-based authentication scheme where trusted identities are provisioned to credit card-sized plastic cards embedded with an Integrated Circuit Chip (ICC). A deployment instance may use multiple authentication use cases depending upon the sensititivity of the various resources that are sought to be protected in its environment. An authentication use case in turn will consist of one or more authentication mechanisms. Every authentication mechanism, in this context, will involve use of the device (the smart card or smart identity token[1] in our context) but some of them may not require participation of the user/bearer of the device since the underlying protocol may not call for the bearer input (e.g., a PIN or biometric sample).

In order to assess the authentication strengths associated with authentication mechanisms using a smart card, we need to look at the typical phases involved in any smart card-based identity verification scheme (smart identity token) deployment. They are:

- Identity Token Eligibility Determination Phase
- Identity Token Issuance Phase
- Identity Token Usage Phase

Out of the three phases above, the authentication mechanisms and by extension the authentication use cases come into the picture only in the Identity Token Usage Phase. Since the objective of this paper is a methodology for assignment of authentication assurance level/strength for authentication use cases, our focus should be on the Identity Token Usage phase. However, we find that in order to arrive at a meaningful authentication strength metric, we need to examine all three phases because of the following rationale.

- The overall authentication strength in authentication use cases deployed in the Identity Token Usage phase is derived from the combination of trust levels in its constitutent authentication mechanisms. The trust level of an authentication mechanism, in turn, is based on the number of trust bindings (embedded in credential objects) it verifies.

- The trust in the set of credential objects that are provisioned to the smart identity token during the Identity Token Issuance phase comes from the bindings it embodies and from the overall security of the system processes used in their generation – security for the data repositories holding the enrollment records, trust in attestion authority that is vouching for credential bindings (e.g., Certificate Authorities (CAs) for digital certificates).

- The basis for creation of credential objects in turn is the “Proofed Identity” which is embodied in the set of data records called enrollment records that are created after a successful “Identity Proofing” process in the Identity Token Eligibility Determination phase.

Thus we see that the trust marker or “Proofed Identity” for the individual being authenticated is established in the Token Eligibility Determination Phase which together with other data in the enrollment records forms the basis for creation of credential objects in the Token Issuance phase. The credential objects by definition embody a “stamp of authority” or trust binding in each of them. Since the purpose of any authentication mechanism is to

[1] We will the two terms interchangeably in this document

verify/validate those bindings, any assessment of its strength should involve the set of bindings it verifies as a prime metric. *Hence identification of the verified bindings of an authentication mechanism logically forms the first step of our methodology*. Before we proceed to that step, we take a look at the various activities leading up to the creation of those credential objects and the issuance of the smart identity token in order to fully understand the nature of the trust chain.

3.1 Trust Creation in Identity Token Eligibility Determination Phase

The primary processes in this phase are identity proofing and enrollment/registration. The “identity proofing” starts with verification of one or more source documents attesting to the identity of the intended card/credential holder together with/without consultation of authoritative data repositories (e.g., use of credit history records and the use of Criminal History database for background verification). The degree of trust in the identity of the individual undergoing identity proofing process is determined by the nature and number of source documents used. This trust is then concretized in an artifact called "proofed identity" in order to be carried over to the next phase of the smart identity token deployment. The most common artifact is usually a set of fingerprints [NSTC2008] which are collected at the conclusion of a successful identity proofing process. This artifact thus creates the “binding” between the person who has undergone identity proofing and the "prospective credential holder/identity token holder" since the tokens are going to carry the provisioned credentials.The biographical details gathered from the source documents together with the proofed identity are stored in a formal system of records (called the enrollment records) during the enrollment/registration process of this phase.

3.2 Creation of Trust Bindings in the Identity Token Issuance Phase

The processes in this phase include the following:

- Assignment of a unique person identifier to the token holder: The person identifier can either be: (a) locally unique (e.g., employee number in an organization) or (b) globally unique (i.e., UUID).

- Creation of credentials that embody various types of “trust bindings” and provisioning them to the token/smart card: The choice of a trust binding and by extension the choice of a credential that embodies that binding is based on the degree of assurance it provides against exploitation of threats EF-T1, EF-T2 and EF-T3 described in section 2. The required assurance, at the minimum, are:

  (a) The assigned person identifier has originated from an authorized credential/token issuer (assurance against the threat of faked or forged credential –FF-T3);

  (b) The assigned person identifier pertains to the person who has been successfully “identity proofed”.It is for this purpose that the “proofed identity” created in the token eligibility determination phase is used (assurance that the token recipient is the person who has undergone “identity proofing”);

  (c) The token instance carrying the person identifier is the physical copy to which the identifier was provisioned by the authorized token issuer (assurance against the threat of cloned token-FF-T2); and

  (d) The person presenting the token (token holder or cardholder) is the person to whom the token was issued by the authorized token issuer. (assurance against the threat of stolen card/impersonation –FF-T1).

- The physical handover of the smart identity token to the legitimate credential owner: Here we need the trust (or assurance) that the person receiving the physical token is the same person for whom identity proofing was done and whose credentials are now provisioned to the token. This assurance is obtained by making the token recipient authenticate against the proofed identity created during the token eligibility phase and now provisioned to the token. For example if a set of fingerprints collected during enrollment is

used as "proofed identity", then the token recipient can be made to authenticate against the same set (or subset) of fingerprints that has been provisioned to the card during card personalization. This is accomplished by the token recipient providing a set of live samples of fingerprints and having a successful match as the pre-condition for receiving the token.

3.3 Verification of Trust Bindings in Identity Token Usage Phase

This phase involves the exercise of one or more of the authentication use cases (designed for the particular smart identity token deployment) by the designated authentication points or stations. As already mentioned, an authentication use case constitutes one or more authentication mechanisms. Each of the authentication mechanisms by design perform the task of verifying one or more trust bindings created during token issuance phase.

Having looked at the processes in the three main phases of a smart identity token deployment, it is now time to have a comprehensive view of the entire lifecycle of processes involved. The primary observation that emanates from taking this viewpoint is that the processes that constitute the Card/Token Issuance phase creates various trust links or bindings using some combination of assigned identity, proofed identity and device-specific secrets which are subsequently verified using various authentication use cases deployed during the identity token usage phase. Hence it follows that a metric for determining authentication strength for any authentication use case should be based on the "set of trust bindings verified" as part of that use case.

The choice of the subset of these bindings that should be verified in any authentication use case depends upon the requirements of the access control application for which smart card-based identity verification in general and the authentication use case in particular is used. More specifically, the requirements are dictated by the value and the sensitivity levels of the resources being protected by the access control application and the impact of wrong identification.

The highest authentication assurance level is provided by those Authentication Use Cases that verify all bindings created during the identity token issuance phase. The use of an Authentication Use Case that provides the highest assurance level cannot be economically justified. Hence any practical smart identity token deployment uses different authentication use cases for different access control applications within the enterprise.

Given the above observations, we are now ready to lay out a roadmap for developing an authentication assurance level methodology for assigning authentication strengths for various smart card-based authentication use cases. Before going into development steps, we want to designate a name for our methodology and an associated abbreviation to refer to it. The abbreviation we have chosen for our methodology is SCIV-ALM that stands for "Smart Card-based Identity Verification - Assurance Level Methodology." The specific steps in our roadmap are the following (we have chosen to denote each step with the abbreviation SCIV-ALM-Tx:

- SCIV-ALM-T1: Identification of the common data objects found in smart identity tokens, the bindings or trust links established/embodied when they were provisioned as part of the issuance process, and primitive authentication mechanism(s) that verifies those bindings during the token usage phase of the smart card deployment.

- SCIV-ALM-T2: Determine the authentication assurance level for each authentication use case used in a real-world smart card-based identity verification deployment based on the primitive authentication mechanisms it comprises of and the number, type and composition of bindings that are verified as a consequence of those mechanisms. Specifically the number and type of verified bindings are used to obtain the "intrinsic authentication strength" of an authentication use case while the composition of the "verified bindings" set is used to derive a partial order (or dominance relationship) among the authentication use cases. These partial orders are used to derive an authentication assurance level taxonomy for the entire deployment instance.

The activities in step SCIV-ALM-T1 are described in the next section whle the corresponding ones in SCIV-ALM-T2 are discussed in Sec. 5.

4. Smart Cards – Common Credential Objects, Embodied Bindings and Verifying Primitive Authentication Mechanisms

So far we have made the case that the methodology for assigning authentication strength for any authentication use case should be based on the set of bindings (created/embodied during the card issuance phase) it verifies. Since an authentication use case is composed of primitive authentication mechanisms, we need to look at the common set of objects on the smart identity token that participate in those mechanisms and the bindings that each of these objects embody. Hence, our next steps involve an analysis of the following – the common set of objects carried in an identity smart card, the binding established/embodied by their provisioning to the card and the primitive authentication mechanism that verifies each of the bindings.

4.1 Objects on Smart Identity Tokens

The following are the common set of electronic objects (as opposed to the visual printed objects) in a smart card used as identity tokens along with their classifiers:

Person Identifiers:

- Digitally signed Cardholder Unique Identifier (CHUID) that carries the Unique Person Identifier appended with the digital signature of the authorized Card/Token Issuer.

Credentials:

- Card Authentication Certificate object (e.g., A digital certificate issued by a CA authorized by the card issuer that attests for the presence of a secret cryptographic key that is exclusive and specific to the copy of the physical token)

- Personal Authentication Certificate object [PKI2008] (e.g., A digital Certificate issued by a CA authorized by the card issuer that binds the Person Identifier of the Cardholder/Credential Holder with a Public Cryptographic key (and through this to a private cryptographic key that is a token-held secret)

- Digitally signed Biometric object (e.g., A biometric record that contains the Person Identifier along with his/her biometric data and is digitally signed by the authorized Card/Token Issuer)

Token-held Secrets:

- Physical Token-exclusive Secret Object (usually carrying the secret associated with an artifact in a public authentication credential - e.g., private key associated with the public key in any digital authentication certificate carried in the Smart Card)

- Secret shared between the Physical Token and the Cardholder (e.g., PIN)

- Secret shared between the Card Issuer and the Physical Token (e.g., A symmetric cryptographic key)

In the following sections, we provide a brief description of each of the above objects, the binding established/embodied when they are provisioned to the smart identity tokens and the primitive authentication mechanism(s) that verify the presence/validity of each of the bindings.

In addition to the above objects, smart identity tokens also contain a class of objects called Security objects. This class of objects is present in a smart identity token to provide integrity checks on other objects (electronic or

printed) found on the card. These objects are generally not used in any authentication use cases. Examples of such objects are:

- The printed information object in a PIV card [PIV2013] that contains the digital representation of the visual/printed objects found on the front and rear of that card.

- The security object that contains the concatenated hash of all the electronic objects found in a PIV card that is digitally signed by the issuer of that card. (to provide integrity checks for the entire electronic content of the card)

- A symmetric cryptography key that is usually used to establish a secure session with the card for the purpose of electronic personalization of the smart card – populating the card with Identifier, Credential or Security objects or for generating other secret objects such as the private key of an asymmetric cryptographic key pair. Thus it carries the binding between the smart card application administrator and the card. Hence, from the cardholder/credential owner point of view, this binding is not considered relevant since the verification of this binding authenticates just authenticates the smart card application administrator.

4.2 Digitally Signed Cardholder Unique Identifier (CHUID) Object

Person Identifier Objects such as the CHUID are those that contain purely identity data usually appended with an artifact that shows their stamp of authority such as a digital signature. The identity data is made up of one or more unique person identifiers (since we are using the smart cards for personal identity verification) accompanied by associated attributes such as the creation date for the identifier, the expiry date for the identifier etc. The uniqueness of these identifiers holds within the domain in which the smart identity tokens are used – such as the unique identifier for every federal government employee (or contractor) in a government smart identity token program (e.g., PIV program of the civilian US Federal government) or a unique identifier for each employee working in port terminals (e.g., Transportation Worker Identity Credential (TWIC) program). The digital signature on the object is generated by the authorized card/token issuer using its private key. The set of characters constituting the digital signature (the signature string) along with the digital certificate containing the associated public key (called the signing certificate) is inserted into the digital signature block that follows the identity data portion of the object. Thus a digitally signed CHUID object establishes the **Card Issuer to Person Identifier binding.**

Based on the purpose of its creation and the stamp of authority it carries, the primitive authentication mechanism using the digitally signed CHUID object involves validation of the unique person identifier on the card. The validation of the unique person identifier consists of the following:

- Ensuring that the person identifier is one of the legitimate identifiers loaded into the access control system (logical or physical); and

- The digital signature of the authorized card issuer over the person identifier object (CHUID) verifies.

The verification of the digital signature in the CHUID object consists of: (a) Establishing trust in the signing certificate (through PKI path validation and certificate status checking) and (b) Verifying that raw signature string was generated by the private key counterpart of the card issuer's public key found on the signing certificate

Theoretically, the unique personal identifier in the CHUID object is considered as the primary source. Hence authentication use cases involving credential objects such as biometric record or personal authentication certificate (that contain the unique person identifier as a component) always perform the extra step of comparing the unique person identifier extracted from these objects (after obtaining assurance in the validity of the credential as a whole) with the unique person identifier found in the digitally signed CHUID object. This comparison also

helps to determine the status of the unique person identifier since attributes associated with the unique person identifier such as expiry date are usually found only in the CHUID object. Thus we see that the person identifier validation is an integral part of any authentication use case involving smart cards since this identifier (along with one or more attributes as needed) is the basis on which all authorization decisions are made.

4.3 Card Authentication Certificate Object

In certain authentication points (stations), especially those that require fast authentication due to high traffic volume, merely demonstrating that the person to be authenticated is in possession of a cryptographic secret attested by the card /token issuer (or any entity authorized to issue credentials on its behalf) is sufficient. The Card Authentication Certificate object is an example of an object that carries such a type of attestation. This certificate object contains the public key counterpart of the private cryptographic key generated and stored in the smart card and hence deemed to be exclusive and specific to the particular copy of the physical token. The subject in this type of certificate is technically the "Token" itself, and hence many smart card deployments do not clearly mandate as to what the value should be for this field. The binding of the Subject with the Public Key is provided by the signature of the certificate issuer who is a CA authorized by the card issuer and hence the Card Authentication Certificate is an example of a token credential object. Therefore, the binding it establishes is the **Card Issuer - Token Secret binding**.

The primitive authentication mechanism verifying the authenticity of the above binding is Card Authentication Certificate validation. This validation process, just like any other digital certificate validation process consists of the following:

(a) Establishing Trust in the Card Authentication Certificate (ensuring that the certificate was issued by a CA authorized by the card issuer and is currently active, and

(b) The verification of signature linking the subject (or subject alternate name) of the certificate and its associated public cryptographic key found on the certificate.

Validating the Card Authentication Certificate (and thus verifying the Card Issuer - Token Secret binding) represents just the first step in verifying that the physical token has been issued by the right trusted authority. A follow-on step that verifies that the attested secret (described in Sec. 4.6) is indeed held inside the smart card is required. This naturally involves testing for the presence of the private cryptographic key that is the counterpart of the public key in the Card Authentication Certificate. It is this step that verifies the **Token Secret - Physical Token binding** and proves that the presented copy of the physical token is indeed the authenticator.

4.4 Personal Authentication Certificate Object

Another common credential object found on a smart token is the Personal Authentication Certificate. This certificate contains the Person Identifier of the Cardholder/Credential Holder (in the Subject or the Subject Alternate Name field) and a Public Cryptographic key counterpart of a token-held secret (i.e., a private cryptographic key). Naturally, the digital signature of the CA that issued this certificate provides the binding between the two values. Therefore the Personal Authentication Certificate establishes the **Person Identifier - Token secret binding**.

Just as in the case of verifying the Card Issuer - Token Secret binding, the primary authentication mechanism for verifying the above binding consists of validating the Digital Authentication Certificate. Again as in the case of Card Issuer- Token Secret binding, the follow-on activity for verifying the Token Secret - Physical Token binding (which thus verifies the secret associated with Person Identifier) involves testing the presence of the private key counterpart (described in Sec. 4.6) of the public key of the certificate, which in this instance is the Personal Authentication Certificate.

4.5 Digitally Signed Biometric Record Object

This is a credential object since it associates the Person Identifier of the Cardholder/Credential Holder with a representation of his/her personal trait (e.g., biometric template) using the digital signature of the card issuer. The resultant digitally signed biometric record object also includes the digital certificate of the signer (generally the card issuer who also signs the CHUID object). Thus a digitally signed biometric record object creates the binding between the person identifier and a personal trait of the cardholder (e.g., his/her fingerprint). Therefore, the primary authentication mechanism using this object involves comparison of a live biometric sample provided by the cardholder to the template that is part of the stored biometric object. Successful execution of this authentication protocol thus verifies the **Person Identifier - Cardholder binding**. The strength of this binding depends upon the process used in obtaining a live sample of the trait from the cardholder for comparison with the digital representation of the trait stored on the token. Obtaining this live sample under the supervision of a human expert ensures the "liveness" property of the sample and provides the necessary authentication strength for the overall authentication mechanism.

4.6 Physical Token-exclusive Secret Object

Devices that are susceptible to their contents being cloned (such as a smart card) are often made to generate and store a secret that cannot be read through any interface (contact or contactless) and copied to a different instance of the physical token.This is usually the private key of an asymmetric key pair which is generated by a cryptographic key processor resident in the card. The processor is programmed to generate a key pair and return only the public portion of the asymmetric key pair while storing the private portion in a tamper proof way inside the card.

The private cryptographic key, though cannot be read through any interface, its presence can be revealed by the card through a "private key operation" on a random challenge sent by an authentication system. The presence of the private cryptographic key, an example of a non-shared secret object, is thus verified. This authentication protocol thus verifies the **Token Secret - Physical Token binding**. The verification of the binding thus addresses the threat due to cloning of a physical artifact carrying the trusted credentials. It does not address the threat of a legitimate card being stolen and used by a non-owner or an impersonator

The private cryptographic key, whose presence is verified, can be the counterpart of a public key in two different types of certificates depending upon the type of authentication. These two types are described below:

4.6.1 Authenticating the Physical Token

In authentication scenarios where the presence of private cryptographic key is verified merely to authenticate the physical token (to ensure that it is the original physical copy of the token issued by the trusted card issuer and not a cloned version), its public key is part of the "Card Authentication Certificate" (Sec. 4.3). The private cryptographic key that is held in the token is therefore called the "Card Authentication Key". Since the purpose here is to merely authenticate the physical token, the card authentication certificate usually carries only a dummy or pseudo identifier in its "Subject" field or "Subject Alternate Name" field.

4.6.2 Authenticating the Association of Person Identifier to the Physical Token

In authentication scenarios where the presence of private cryptographic key is verified because it is treated as a "Secret" associated the person identifier, its public key counterpart is part of the "Personal Authentication Certificate" (Sec. 4.4). The private cryptographic key that is held in the token is therefore called the "Personal Authentication Key." Since the purpose here is to verify the association of the person identifier with the physical token, the personal authentication certificate will have in its "Subject" or "Subject Alternate Name" field, the person identifier as its value.

4.7 Secret Shared Betwen Physical Token and Cardholder

The verification of the Person Identifier - Token Secret binding (established by Personal Authentication Certificate) and the verification of the Token Secret - Physical Token binding (established by the non-shared secret object, i.e., Private Cryptographic Key) involves mechanisms that does not require cardholder participation and hence authentication use cases that involve these mechanisms will be successful even in situations where the cardholder is an imposter (i.e., not the card owner/credential holder). Thus the presence of the Personal Authentication Certificate Object and the Physical Token-exclusive secret object (Sec. 4.6) are not sufficient for providing assurance against a stolen card.

To provide this assurance, an object that is of the nature of a shared secret and that establishes the binding between the physical token and the cardholder must be created in the smart identity token. Verification of this binding then provides the assurance that the cardholder is indeed the legitimate owner of the token/credentials. Fortunately, the smart card technology provides the capability to define a “PIN” object whose verification can be made as an access control condition for reading certain "sensitive objects" or for performing certain “sensitive operation.” Using this feature, the identity verification application on the card can be configured to have the “PIN Verification” as an access control condition for the “private key operation” (i.e., digitally signing a random challenge sent by the authentication system using a private cryptographic key), which is one of integral operations of the protocol for testing the presence of a private cryptographic key on the card. Making the card recipient choose a PIN value at the time of issuance in the presence of the card issuer thus completes the process of establishing the **Physical Token - Cardholder binding**. The primitive authentication mechanism of the cardholder presenting the PIN during the time of authentication verifies this binding and thus addresses the threat of an imposter presenting a stolen card.

4.8 Secret Shared Between Card Issuer and Physical Token

In some instances, the combination of Card Issuer - Token Secret and Token Secret - Physical Token bindings are established by the Card Issuer by injecting a secret symmetric cryptographic key into the smart identity token at the time of issuance. The verification of these bindings takes place through a protocol that enables the token (and in some instances the authentication system as well) to reveal the presence of this symmetric cryptographic key. The primitive authentication mechanism incorporating this protocol is not widely deployed since it requires the authentication station be in possession of the entire set of symmetric keys for all the smart tokens that will be used at that location. Further, only authentication systems in the native domain of the card issuer can be trusted to hold these symmetric secret keys and hence cannot be used in federated and widely interoperable smart identity token deployments.

The common set of smart card objects, the bindings established by each of them and the primitive authentication mechanisms that verifies each of those bindings that are discussed in Sec. 4.2 through Sec. 4.8 above are summarized in Table 1 below.

Table 1: Smart Card Objects, Bindings and Verifying Authentication Mechanisms

Card Object Name - Description	**Binding Established/Embodied**	**Primitive Authentication Mechanism (verifying the bindings)**
Card Holder Unique Identifier (CHUID) Object- An Object containing the Unique Person Identifier that is usually digitally signed by the Card Issuer	Card Issuer – Person Identifier Binding (Strong)	**PUM-1: Person Identifier’s origin and integrity** checked using its associated digital signature
Card Authentication Certificate	Card Issuer – Token Secret	**PUM-2.1**: Establishing Trust in

Object – A digital certificate issued by a CA trusted by Card Issuer that attests to the presence of a cryptographic secret held by the token.	Binding (Strong) (OR) Person Identifier – Token Secret Binding (Strong) (if the card authentication certificate contains the person identifier)	the Card Authentication Certificate (ensuring that the certificate was issued by a trusted, authorized CA, is currently active and that the digital signature generated by the certificate issuer verifies)
Personal Authentication Certificate Object – A digital certificate issued by a CA trusted by Card Issuer that attests to the association of the person identifier with a cryptographic secret held by the token.	Person Identifier – Token Secret Binding (Strong)	**PUM-2.2**: Establishing Trust in the Personal Authentication Certificate (ensuring that the certificate was issued by a trusted, authorized CA, is currently active and that the digital signature generated by the certificate issuer verifies)
An Object containing Physical Token-exclusive Secret - Verifying Presence of non-shared embedded token secret (e.g., An asymmetric private cryptographic key)	Token Secret – Physical Token Binding (Strong)	**PUM-3**: Verifying Presence of non-shared embedded token secret (tested by sending an input data from the Verifier and verifying the token response through a related artifact) (e.g., checking the presence of asymmetric private key by sending a random challenge and verifying the signed response using its associated validated public key)
An Object containing a Secret shared between the Physical Token and the Cardholder (chosen by the recipient of the card at the time of issuance)	Physical Token – Cardholder Binding (Strong or Weak depending upon the entropy of the shared secret)	**PUM-4**: Verifying the presence of a secret shared (e.g., PIN) between Physical Token and Cardholder (usually used as an access control mechanism for an authentication protocol (e.g., PUM-3))
An Object containing a Secret shared between the Card Issuer and the Physical Token (e.g., a symmetric cryptographic key injected into the smart card by the issuing system at the time of card issuance)	Card Issuer – Physical Token Binding (Strong)	**PUM-5**: Verifying the presence of a secret shared between the Card Issuer and the Physical Token (e.g., a symmetric cryptographic key)
Biometric Record Object: Credential object linking the person identifier with the data representing the personal trait of the cardholder using the digital signature of the card issuer	Person Identifier – Cardholder Trait Binding (Strong)	**PUM-6**: Trust in the signing certificate established through Certificate Validation and the digital signature in the signed biometric object verified

	Cardholder Trait – Cardholder (Strong or Weak depending upon the process used in obtaining a live sample of the trait for comparison with the digital representation of the trait stored on the token)	**PUM-7.1**: The cardholder presents a live sample of biometric data in an unattended authentication station (OR) **PUM-7.2**: The cardholder presents a live sample of biometric data in an attended authentication station

4.9 Threat Coverage of Primitive Authentication Mechanisms

Primitive authentication mechanisms that perform the task of either: (a) validating the attestation of the person identifier by the authorized card/token issuer by verifying the digital signature attached it (PUM-1) (or) (b) verifying the binding of the person identifier with a private cryptographic secret by first validating the certificate that contains its public counterpart and the person identifier and then testing the presence of the private cryptographic secret (PUM-2.2 and PUM-3) address the threat of forged credentials (FF-T3).

Primitive authentication mechanisms that perform the task of either: (a) verifying the binding of the person identifier with a personal trait sample (e.g., biometric) and perform a comparison of a live sample of a biometric from a cardholder with the validated trait stored on the card (PUM-6, PUM-7 and PUM-8), or (b) verifying cardholder knowledge of a secret shared between the himself/herself (PUM-4) address the threat of an imposter presenting somebody else's token to the authentication system (FF-T1).

Primitive authentication mechanisms that perform the task of either: (a) validating a certificate object that contains an attestation of a non-tamper proof secret specific to the particular physical copy of the token and then testing the presence of that tamper-proof cryptographic secret (PUM-2.1 and PUM-3) or (b) testing the presence of a cryptographic secret shared between the card issuer and the particular physical copy of the token (PUM-5) address the threat of a cloned copy of a valid token being presented to the authentication system (FF-T2).

Thus we see that the set of all primitive authentication mechanisms collectively provide coverage for the entire set of threats identified in Sec. 2. It must be mentioned however that some of the primitive authentication mechanisms address a threat only partially (we use 50% value to denote this) and need a follow-on mechanism (that together form a complete authentication protocol) to completely address a threat. The threats addressed (and not addressed) by each primitive authentication mechanism are listed in Table 2 below:

Table 2: Threat Coverage of Primitive Authentication Mechanisms

Primitive Authentication Mechanism (*Bindings Verified*)	**Threats Addressed (*Percentage of Coverage)***	**Unaddressed Threat (and hence) Inherent Weakness**
PUM-1: Person Identifier's origin and integrity checked using its associated digital signature (*Card Issuer – Person Identifier Binding (Strong))*	FORGED CREDENTIAL (*100%*)	1.STOLEN CARD – cardholder non-participation 2. CLONED CARD – physical token not validated
PUM-2.1: Establishing Trust in the Card Authentication Certificate (ensuring that the certificate was issued by a trusted, authorized CA, is currently active and that the digital signature	CLONED CARD (*50%*)	Validity of Person Identifier is established (by linking to a legitimate token-held secret) STOLEN CARD – cardholder

generated by the certificate issuer verifies) (Card Issuer – Token Secret Binding (Strong))		non-participation
PUM-2.2: Establishing Trust in the Personal Authentication Certificate (ensuring that the certificate was issued by a trusted, authorized CA, is currently active and that the digital signature generated by the certificate issuer verifies) (Person Identifier – Token Secret Binding (Strong))	FORGED CREDENTIAL (*100%*) CLONED CARD (*50%*)	
PUM-3: Verifying Presence of non-shared embedded token secret (tested by sending an input data from the Verifier and verifying the token response through a related artifact) (e.g., checking the presence of asymmetric private key by verifying the signed response using its associated validated public key) (Token Secret – Physical Token Binding (Strong))	CLONED CARD (*50%*)	Validity of the Physical Token is established STOLEN CARD – cardholder non-participation
PUM-4: Verifying the presence of a secret shared (e.g., PIN) between Physical Token and Cardholder (usually used as an access control mechanism for an authentication protocol (e.g., PUM-3)) (Physical Token – Cardholder Binding (Strong or Weak depending upon the entropy of the shared secret))	STOLEN CARD (*100%*)	Card Holder is authenticated to Physical Token CLONED CARD – association between physical token and cardholder can be established without card issuer presence FORGED IDENTIFIER – identifier non-participation
PUM-5: Verifying the presence of a secret shared between the Card Issuer and the Physical Token (e.g., a symmetric cryptographic key) (Card Issuer – Physical Token Binding)	CLONED CARD(*100%*)	Validity of the Physical Token is established STOLEN CARD – cardholder non-participation FORGED IDENTIFIER – identifier non-participation
PUM-6: Trust in the signing certificate established through Certificate Validation and the digital signature in the signed biometric object verified (Person Identifier - Cardholder Binding (Strong))	FORGED CREDENTIAL (*100%*) STOLEN CARD (*50%*)	Validity of Biometric Object Established (by linking it to an authorized creator/card issuer) CLONED CARD – physical token not validated
PUM-7.1: The cardholder presents a live sample of biometric data in an unattended authentication	STOLEN CARD (*50%*)	Cardholder is weakly authenticated (by linking to the validated biometric object on the

station (Cardholder Trait – Cardholder Binding (Weak))		card) CLONED CARD – physical token not validated
PUM-7.2: The cardholder presents a live sample of biometric data in an attended authentication station (Cardholder Trait – Cardholder Binding (Strong))	STOLEN CARD (*50%)*	Cardholder is strongly authenticated (by linking to the validated biometric object on the card) CLONED CARD – physical token not validated

5. Development of Authentication Assurance Level Taxonomy for Canonical Authentication Use Cases

In the previous section, we identified the common data objects in a smart identity token, the trust binding each embodied and the primitive authentication mechanisms that verified those bindings. These activities completed all the activities for the first step SCIV-ALM-T1 of our authentication assurance methodology. The next step of our methodology SCIV-ALM-T2 uses the data obtained from SCIV-ALM-T1 to assign authentication assurance levels to various authentication use cases specified in a smart identity token deployment. An authentication assurance level in our methodology is made up of two components – the intrinsic authentication strength and relative authentication strength. The process used in deriving these two components constitutes the activities of the current step SCIV-ALM-T2.

- For each authentication use case specified for a deployment, identify the set of primitive authentication mechanisms it is made up of and the consequent number, type and composition of verified bindings.

- Use the number and type of verified bindings to obtain the "intrinsic authentication strength" of an authentication use case.

- Use the composition of the "verified bindings" set and the intrinsic authentication strength of each authentication use case as metrics to derive a partial order (or dominance relationship) sequences among the authentication use cases and use these partial order sequences to develop an authentication assurance level taxonomy for the entire deployment instance.

A close examination of the above two activities may give one the impression that in order to develop our authentication assurance methodology, we need some practical authentication use cases from some real-world smart identity token deployments. In other words, it sounds like we need proof of concept as an integral part of methodology development. In order to avoid that conundrum, we demonstrate the development of rest of our methodology using a set of authentication use cases developed from some first principles. These first principles dictate that the basic goal of any authentication use case is to obtain assurance against one or more of the threats in the set EF-T1, EF-T2 and EF-T3 (refer Sec. 2). To realize this basic goal, every authentication use case must be made up of fundamental building blocks called "Assurance Elements". Hence it follows that each assurance element must address at least one of the threats in the set referred above. Based upon this logic, we proceed to identify the set of basic assurance elements needed for any smart card-based identity verification deployment.These assurance elements along with threats addressed (in parenthesis) are:

- A-E1: The credentials on the smart identity token presented to the authentication system have originated from an authorized credential/card issuer and has not been tampered with after issuance – *Assurance of non-forged Credentials (EF-T3)*

- A-E2: The physical copy of the token presented to the authentication system is the instance to which the credentials were provisioned by the card issuer – *Assurance against Cloned Card (EF-T2)*

- A-E3: It can be verified that the credentials do belong to the person presenting the identity token – *Assurance against Stolen Card (EF-T1)*

Since each of the above assurance elements cannot be broken into any sub-elements, we can treat each of them as elements of a basis vector, which we shall call as "Assurance Basis Vector." We can then use all allowable subsets of elements that constitute the assurance basis vector to derive what we term as canonical authentication use cases. The criterion for an allowed subset is that it should include "non-forged credential (personal identifier)" as an assurance element. This is justified based on the fact that irrespective of the sophistication of the authentication protocol, it is the "authenticated person identifier" extracted from the smart card that is going to be the basis for any access (physical or logical) control decision.Our rationale for terming the allowed subsets as "canonical" is based on the fact that they are derived from the basic set of assurance elements.

The set of canonical authentication use cases (along with an identifier for each), the constitutent assurance elements in each of them together with a brief rationale for its possible choice as a candidate in practical smart card-based identity verification deployments is given in Table 3 below:

Table 3: Canonical Authentication Use Cases

Canonical Authentication Use Case	Constituent Assurance Element(s)	Rationale for Use in Real-World Deployments
Authenticate the Credential (A-UC1)	*Assurance of non-forged Credentials (Person Identifier) (A-E1)*	For low-security applications such as entry into a conference room for an authorized employee within the building
Authenticate the Credential and Authenticate the Physical Token (A-UC2)	*Assurance against Cloned Card (A-E2)+ Assurance of non-forged Credentials (A-E1)*	A large population of users is using the card and there is high probability of cards stolen and/or unreported card loss
Authenticate the Credential and Authenticate the Cardholder (A-UC3)	*Assurance against Stolen Card (A-E3)+ Assurance of non-forged Credentials (A-E1)*	A special set of credentials is issued to a controlled population of users and any proliferation of credentials through cloned tokens is a security risk
Authenticate the Credential, Authenticate the Cardholder (A-UC2) and Authenticate the Physical Token (A-UC4)	*Assurance against Cloned Card (A-E2)+ Assurance against Stolen Card (A-E3) + Assurance of non-forged Credentials (A-E1)*	A high security security environment where credential proliferation as well as stolen cards are security risks

The next step in our methodology is to identify the set of constitutent primitive authentication mechanisms that will provide the assurance referred to in each assurance element of the assurance basis vector. For this, we match the threat addressed by each assurance element (obtained from the definition above) with the "Threats Addressed" column in Table 2. The corresponding primitive authentication mechanism and the associated binding verified can then be read out from the first column of the same table. It can be seen from Table 2 that some primitive authentication mechanisms address a particular threat only partially (always assigned a value 50 %). In these situations, we look for a follow-on primitive authentication mechanism that will complete the authentication protocol for fully addressing that threat. This process is repeated for every authentication assurance element.

At this stage, we have obtained the set of primitive authentication mechanism(s) (with their associated verified bindings) for each assurance element. Since the canonical authentication use cases are nothing

but combinations involving one or more assurance elements, we add up the constitutent primitive mechanisms and the verified bindings for each constitutent assurance element to obtain these values for it.

The set of primitive authentication mechanisms and the set of verified bindings for each canonical authentication use case constructed using data in Table 2 is given in Table 4 below:

Table 4: Verified Bindings (along with their types) for Canonical Authentication Use Cases

Canonical Authentication Use Case	Primitive Authentication Mechanism(s)	Verified Bindings and Their Associated Type (Strong or Weak)
Authenticate the Credential (A-UC1)	**PUM-1**: Person Identifier's origin and integrity checked using its associated digital signature	*1.* (*Card Issuer – Person Identifier Binding (Strong))*
Authenticate the Credential and Authenticate the Physical Token (A-UC2)	**PUM-1**: Person Identifier's origin and integrity checked using its associated digital signature **PUM-2.1(or PUM-2.2)**: Establishing Trust in the Card Authentication Certificate (or Personal Authentication Certificate) (ensuring that the certificate was issued by a trusted, authorized CA, is currently active and that the digital signature generated by the certificate issuer verifies) **PUM-3**: Verifying Presence of non-shared embedded token secret (tested by sending an input data from the Verifier and verifying the token response through a related artifact) (e.g., checking the presence of asymmetric private key by verifying the signed response using its associated validated public key)	*1.* (*Card Issuer – Person Identifier Binding (Strong))* *2. (Card Issuer – Token Secret Binding (Strong))* *3. (Token Secret – Physical Token Binding (Strong))*
Authenticate the Credential and Authenticate the Cardholder (A-UC3)	**PUM-1**: Person Identifier's origin and integrity checked using its associated digital signature **PUM-4**: Verifying the presence of a secret shared (e.g., PIN) between the cardholder and the physical token. (usually used as an access control mechanism for another authentication protocol (e.g., PUM-3))	*1.* (*Card Issuer – Person Identifier Binding (Strong))* *2. (Physical Token – Cardholder Binding (Strong or Weak depending upon the entropy of the shared secret))*
Authenticate the Credential, Authenticate the Physical Token and Authenticate the Cardholder (A-UC4)	**PUM-1**: Person Identifier's origin and integrity checked using its associated digital signature	*1.* (*Card Issuer – Person Identifier Binding (Strong))* *2. (Card Issuer – Token Secret Binding (Strong))*

	PUM-2.1(or PUM-2.2): Establishing Trust in the Card Authentication Certificate (or Personal Authentication Certificate) (ensuring that the certificate was issued by a trusted, authorized CA, is currently active and that the digital signature generated by the certificate issuer verifies) **PUM-3**: Verifying Presence of non-shared embedded token secret (tested by sending an input data from the Verifier and verifying the token response through a related artifact) (e.g., checking the presence of asymmetric private key by verifying the signed response using its associated validated public key) **PUM-4**: Verifying the presence of a secret shared (e.g., PIN) between the cardholder and the physical token. (usually used as an access control mechanism for another authentication protocol (e.g., PUM-3))	*3. (Token Secret – Physical Token Binding (Strong))* *4. (Physical Token – Cardholder Binding (Strong or Weak depending upon the entropy of the shared secret))*

Now that we have the number and type of verified bindings for each authentication use case, we are now ready to assign the "intrinsic authentication strength" for each canonical authentication use case. We use two symbols to denote the intrinsic authentication strength. They are:

- the number of verified bindings (e.g., for A-UC2 , this number is 3 since three bindings are verified); and
- a negative sign (-) to denote each binding whose verification is weak. (e.g., for A-UC3, one of the binding verifications may be weak and hence a single negative sign is used).

To provide some examples – we find that the canonical authentication use case A-UC2 consists of two binding verifications and since both are strong, the intrinsic authentication strength symbol for A-UC2 is [3]. On the other hand, for A-UC3, one of the verified bindings of the total two may be weak and hence the intrinsic authentication strength for A-UC3 is denoted as [2] (if the second binding is verified using a strong mechanism) or [2-] (if the second binding is verified using a weak mechanism). The intrinsic authentication strength for Canonical Authentication Use Cases is provided in Table 5 below:

Table 5: Intrinsic Authentication Strength for Canonical Authentication Use Cases

Canonical Authentication Use Case	Intrinsic Strength
Authenticate the Credential (A-UC1)	[1]
Authenticate the Credential and Authenticate the Physical Token (A-UC2)	[3]

Authenticate the Credential and Authenticate the Cardholder (A-UC3)	[3 -]
Authenticate the Credential, Authenticate the Physical Token and Authenticate the Cardholder (A-UC4)	[4] or [4-]

The next activity in our authentication assurance level assignment process is identification of partial orders among canonical authentication use cases based on the composition of verified bindings and intrinsic authentication strengths as metrics. From the data on the third column of Table 4, we can identify two partial orders: (A-UC1, A-UC2, A-UC4) and (A-UC1, A-UC3, A-UC4). Looking at the partial orders we find that all of them have the lowest element common. Hence we can use this element (the authentication use case A-UC1 – Authenticate the Credential) as the root node of the taxonomy structure. The resulting taxonomy diagram consisting all four canonical authentication use cases as nodes along with their intrinsic authentication strengths is shown in Fig. 1. It is interesting to note that based on the composition of the verified bindings, the resulting authentication assurance taxonomy has a lattice structure instead of a linear structure which would have resulted if we had merely considered just the number of verified bindings in each authentication use case.

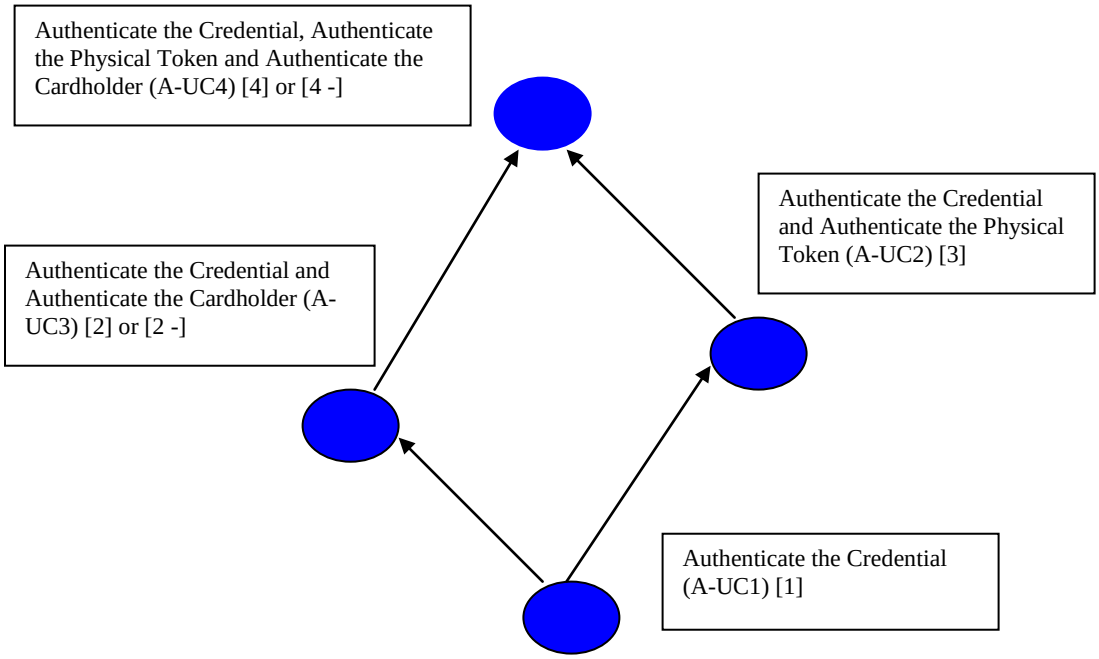

Figure 1: Authentication Assurance Level Taxonomy for Canonical Authentication Use Cases

To apply our methodology to derive authentication strengths and by extension an authentication assurance taxonomy for the set of authentication use cases in a real-world smart identity token deployment, we adopted the following approach: We first derive a table similar to Table 4 except that the first column will contain the authentication use cases designated in the deployment specification. We then examine the constituent protocols in each authentication use case in the deployment specification and map those

protocols to our primitive authentication mechanisms and their associated bindings and thus derive the content for second and third columns of Table 4 for the deployment instance. Once the Table 4 data that contains authentication strengths for authentication use cases in the deployment specification is compiled, the process for deriving the associated authentication assurance taxonomy for the deployment authentication use case set is identical to what has already described in the methodology.

The following are two possible approaches for deriving the set of primitive authentication mechanisms for each authentication use case in a deployment specification. They are:

- Express a deployment authentication use case in terms of canonical authentication use cases and use Table 3 to obtain the constituent assurance elements. The "threat addressed" by each assurance element is read off from the definition of assurance elements (A-E1 through A-E3). The set of primitive authentication mechanisms (and their associated verified bindings) that addresses the set of threats is then obtained from Table 2; and

- Express a deployment authentication use case directly in terms primitive authentication mechanisms and use the data in Table 2 to obtain the set of associated verified bindings

6. Conclusions and Benefits

Examination of the nature of some authentication protocols in smart card-based identity verification deployments together with threats associated with form factor of the devices convinced us that the authentication factor-based approach does not provide us a true measure of authentication strengths and that an alternate methodology is required. Towards this goal, we looked at the typical processes involved in any smart card-based identity verification scheme and found that the most common credential objects populated on the card embodied some form of trust link or binding involving a pair of the following objects - the physical token, the token secret, the cardholder trait (e.g., biometric), card issuer and the person identifier of the card/credential owner. We also observed that any authentication mechanism using these objects involves verification of these embedded pair-wise bindings and that it is the composition of the bindings verified that should provide an intrinsic measure of the authentication strength of these mechanisms. Since an authentication use case is nothing but a combination of authentication mechanisms, it is a straightforward exercise to compute their authentication strength knowing the strengths of the constituent authentication mechanisms. We also showed that the composition of verified bindings can also be used to derive an authentication assurance taxonomy for the entire set of authentication use cases specified for the deployment. The advantages of the bindings-based approach for the measuring authentication strengths are: (a) It is a direct measure of the trust links embedded in the various credential objects embedded in the smart card and (b) the verification of the bindings directly provide coverage for some of the technology-specific vulnerabilities (e.g., stolen card, cloned card etc).

Bibliography

[EAG2013] W. E. Burr, D. F. Dodson, E. M. Newton, R. A. Perlner, W. T. Polk, S. Gupta, and E. A. Nabbus, *Electronic Authentication Guideline*, NIST Special Publication (SP) 800-63-2, National Institute of Standards and Technology, Gaithersburg, Maryland, August 2013. http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-63-2.pdf

[Ham2001] E.-M. Hamann, H. Henn, T. Schäck, and F. Seliger, "Securing e-business applications using smart cards," *IBM Systems Journal* 40(3), 635-647 (October 2001). http://dx.doi.org/10.1147/sj.403.0635

[HSPD2004] Homeland Security Presidential Directive 12: Policy for a Common Identification Standard for Federal Employees and Contractors, Aug 27, 2004. http://www.dhs.gov/homeland-security-presidential-directive-12

[Kum2008] M. Kumar, "New remote user authentication scheme using smart cards," *IEEE Transactions on Consumer Electronics* 50(2) 597 – 600 (May 2004). http://dx.doi.org/10.1109/TCE.2004.1309433

[NSTC2008] National Science and Technology Council (NSTC). Subcommittee on Biometrics and Identity Management, *Identity Management Task Force Report 2008*, September 2008, 216pp. http://www.biometrics.gov/documents/idmreport_22sep08_final.pdf

[OMB2003] Office of Management and Budget (OMB), *E-Authentication Guidance for Federal Agencies*, OMB Memorandum 04-04, December 16, 2003. http://www.whitehouse.gov/sites/default/files/omb/assets/omb/memoranda/fy04/m04-04.pdf

[PIV2013] U.S. Department of Commerce, *Personal Identity Verification (PIV) of Federal Employees and Contractors*, Federal Information Processing Standards (FIPS) Publication 201-2, August 2013. http://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.201-2.pdf

[PKI2008] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk, *Internet X.509 Public Key Infrastructure Certification and Certificate Revocation List (CRL) Profile*, Internet Engineering Task Force (IETF) Network Working Group Request for Comments (RFC) 5280, May 2008 http://www.ietf.org/rfc/rfc5280.txt [accessed 2/20/14].

[TWIC2008] U.S. Department of Homeland Security. Transportation Security Administration, *TWIC Reader Hardware and Card Application Specification*, Version 1.1.1, May 30, 2008. http://www.tsa.gov/sites/default/files/publications/pdf/twic/twicreaderhardwareandcardapplicationspecification.pdf

Appendix A—Case Study: PIV Authentication Use Cases

In this Appendix, we illustrate the application of SCIV-ALM methodology to U.S.Government's identity verification program –the Personal Identity Verification (PIV). We present the case study of the application of SCIV-ALM to PIV program's authentication use cases using the following steps:

- General overview of the PIV program
- Brief description of PIV Authentication Use Cases and their assigned assurance levels in PIV's specification document FIPS 201-2 [PIV2013].
- The assignment of intrinsic authentication strengths to PIV Authentication Use Cases using SCIV-ALM.
- The SCIV-ALM Authentication Assurance Taxonomy for PIV Authentication Use Cases
- Comparison of authentication assurance levels assigned in the PIV specification with the authentication assurance levels assigned by SCIV-ALM.

A.1 Overview of PIV Program

The PIV program is the outcome of the Homeland Security Presidential Directive 12 [HSPD2004] dated August 27, 2004 which directed the promulgation of a Federal Standard for secure and reliable forms of identification for Federal employees and contractors. The overall goal of the program is to achieve appropriate security assurance for multiple applications by efficiently verifying the claimed identity of individuals seeking physical access to federally controlled government facilities and electronic access to government information systems.The core specifications for the PIV program are described in the Federal Information Processing Standards document FIPS 201 and its revisions. The Authentication Use Cases used for our methodology demonstration comes from this document.

A.2 Brief Description of PIV Authentication Use Cases & Specified Assurance Levels

In the context of the PIV Card Application, identity authentication is defined as the process of establishing confidence in the identity of the cardholder presenting the PIV card. Hence the purpose of a PIV Authentication Use Case is to obtain a particular level of assurance that the holder of the PIV card is the owner of the card, depending upon the specific PIV data used to authenticate the holder of the PIV card. Based on this logic, the the Authentication Assurance Levels chosen in FIPS 201 standard are in given Table A.1 below:

Table A.1. PIV Authentication Assurance Levels

PIV Assurance Level	Description
Little or NO Confidence	Little or no assurance in the identity of the cardholder
SOME Confidence	A basic degree of assurance in the identity of the cardholder
HIGH Confidence	A strong degree of assurance in the identity of the cardholder
VERY HIGH Confidence	A very strong degree of assurance in the identity of the cardholder

The PIV Authentication Use Cases and their assigned Assurance Levels are given in Table A.2 below:

Table A.2. PIV Authentication Use Cases and Assurance Levels (Logical Access – Local Work Station)

PIV Authentication Use Case	Brief Description	PIV Assurance Level
6.2.5 – CHUID -Authentication Using the CHUID	The digital signature of the CHUID is verified to ensure that it was issued by the right authority and has not been altered or tampered. The expiration date on the CHUID is also checked. The unique Identifier in CHUID (i.e., FASC-N) is used as an input to Authorization system	LITTLE or NO Confidence
6.2.3.2 – PKI-CAK - Authentication with the Card Authentication Certificate	The Card Authentication Certificate is read and Validated. The signed response to a random challenge is verified using the public key of the certificate. The Unique Identifier in the Certificate is used as an input to Authorization system	SOME Confidence
6.2.1.1 – BIO- Unattended Authentication Using PIV Biometric	The CHUID object is read and its digital signature verified. The biometric object is read after PIN verification. The signing certificate associated with signed biometric object is validated and the digital signature is verified. The cardholder presents a live biometric sample in an unattended authentication station	HIGH Confidence
6.2.1.2 – BIO-A- Attended Authentication of PIV Biometric	The CHUID object is read and its digital signature verified. The biometric object is read after PIN verification. The signing certificate associated with signed biometric object is validated and the digital signature is verified. The cardholder presents a live biometric sample in the presence of an attendant (e.g., security guard)	VERY HIGH Confidence
6.2.2 – OCC-AUTH - Authentication Using On-Card Biometric Comparison	(It is assumed that a signed biometric object containing the biometric template and the unique identifier exists in the card (though not read)) A live sample of the biometric is	VERY HIGH Confidence

	presented directly on the card.	
6.2.3.1– PKI-AUTH- Authentication with the PIV Authentication Certificate	The Personal Authentication Certificate is read and validated. A PIN is provided for authorizing the private key operation. The signed response to a random challenge is verified using the public key of the certificate. The Unique Identifier in the Certificate is used as an input to Authorization system	VERY HIGH Confidence

A.3 SCIV-ALM Assigned Intrinsic Authentication Strengths for PIV Authentication Use Cases

We now assign intrinsic authentication strengths (in terms of the number and type of verified bindings) to PIV Authentication Use Cases based on SCIV-ALM. In order to accomplish this, we examine the protocols involved in each PIV Authentication Use Case and map them directly to the primitive authentication mechanisms of our methodology. Once the SCIV-ALM primitive authentication mechanisms are identified, the associated "verified bindings" and their type are obtained directly from Column 1 of Table 2 in Sec. 4.9. The constitutent primitive authentication mechanisms and their associated verified bindings (along with their types) for all PIV authentication use cases is shown in Table A.3 below. This approach is possible because of the granularity of description of each PIV Authentication Use Case in the FIPS 201-2 specification document.

In this context, an observation is in order regarding the functionality provided by the primitive authentiation mechanism PUM-1. PUM-1 provides the assurance that the person identifier in the smart token has been issued by the authorized card issuer and has since not been tampered with. It accomplishes this objective by verifying the digital signature associated with a CHUID object that is dedicated for carrying this person identifier value and its associated attributes. However, the same assurance is also provided by PUM-2.1, PUM-2.2 and PUM-6 primitive authentication use cases. This is the underlying logic for including PUM-1 (and its associated verified bindings) as a constitutent primitive authentication mechanism whenever the above three primitive authentication mechanisms (i.e., PUM-2.1, PUM-2.2 and PUM-6) are part of constitutent primitive authentication mechanisms in any PIV authentication use case. This is the case with PIV authentication mechanisms PKI-CAK, BIO, BIO-A and PKI-AUTH and is appropriately shown in Table A.3.

In PUM-2.1 and PUM-2.2, the origin and integrity of the person identifier are ensured by validating the digital certificate that includes this value. In PUM-6, the same assurance is provided by verifying the digital signature of the signed biometric object that contains the person identifier.

Table A.3. SCIV-ALM Verified Bindings for PIV Authentication Use Cases

PIV Authentication Use Case	SCIV-ALM Primitive Authentication Mechanisms involved	SCIV-ALM Verified Bindings (Type)
6.2.5 – CHUID -Authentication Using the CHUID	PUM-1: Person Identifier's origin, status, and integrity checked using its associated digital signature	*Card Issuer – Person Identifier Binding (Strong)*
6.2.3.2 – PKI-CAK - Authentication with the Card Authentication Certificate	PUM-1 functionality by equivalent operations PUM-2.1: Establishing Trust in the Card Authentication Certificate (ensuring that the certificate was issued by a trusted, authorized CA, is currently active and that the digital signature generated by the certificate issuer verifies), The unique person identifier is obtained from the certificate PUM-3: Verifying Presence of non-shared embedded token secret (tested by sending an input data from the Verifier and verifying the token response through a related artifact) (e.g., checking the presence of asymmetric private key by verifying the signed response using its associated validated public key)	*1.* (*Card Issuer – Person Identifier Binding (Strong)) (Implicit)* *2. (Person Identifier – Token Secret Binding (Strong))* *3. (Token Secret – Physical Token Binding (Strong))*
6.2.1.1 – BIO- Unattended Authentication Using PIV Biometric	PUM-1 functionality by equivalent operations PUM-6: The signed biometric object is read after meeting the ACL condition using a PIN. Trust in the signing certificate established through Certificate Validation and the digital signature in the signed biometric object verified PUM-7: The cardholder presents a live sample of biometric data in an unattended authentication station	*1. Card Issuer – Person Identifier Binding (Strong)* *2. Person Identifier – Cardholder Trait Binding (Strong)* *3. Cardholder Trait – Cardholder Binding (Weak)* *4. Physical Token – Cardholder Binding (Strong or Weak depending upon PIN size)*
6.2.1.2 – BIO-A- Attended Authentication of PIV Biometric	PUM-1 functionality by equivalent operations PUM-6: The signed biometric object is read after meeting the ACL condition using a PIN. Trust in the signing certificate established through Certificate Validation and the digital signature in the signed biometric	*1. Card Issuer – Person Identifier Binding (Strong)* *2. Person Identifier – Cardholder Trait Binding (Strong)* *3. Cardholder Trait – Cardholder Binding (Strong)* *4. Physical Token – Cardholder Binding (Strong or Weak depending upon PIN size)*

	object verified PUM-8: The cardholder presents a live sample of biometric data under the supervision of an attendant (e.g., Security Guard)	
6.2.2 – OCC-AUTH - Authentication Using On-Card Biometric Comparison	(It is assumed that a signed biometric object containing the biometric template and the unique identifier exists in the card (though not read)) A live sample of the biometric is presented directly on the card.	*1. Card Issuer – Person Identifier Binding (Strong)* *2. Person Identifier – Cardholder Binding (Strong or Weak depending on whether there is an attendant or not)*
6.2.3.1– PKI-AUTH- Authentication with the PIV Authentication Certificate	PUM-1 functionality by equivalent operations PUM-2.2: Establishing Trust in the Personal Authentication Certificate (ensuring that the certificate was issued by a trusted, authorized CA, is currently active and that the digital signature generated by the certificate issuer verifies) The unique person identifier is obtained from the certificate PUM-3: Verifying Presence of non-shared embedded token secret (tested by sending an input data from the Verifier and verifying the token response through a related artifact) (e.g., checking the presence of asymmetric private key by verifying the signed response using its associated validated public key) PUM-4: Verifying the presence of a secret shared (e.g., PIN) between the cardholder and the physical token. (usually used as an access control mechanism for another authentication protocol (e.g., PUM-3))	*1. Card Issuer – Person Identifier Binding (Strong) (Implicit)* *2. Person Identifier – Token Secret Binding (Strong)* *3. Token Secret – Physical Token Binding (Strong)* *4. Physical Token – Cardholder Binding (Strong or Weak depending upon PIN size)*

Using the number and type of verified bindings for each PIV Authentication Use Case, we assign the intrinsic authentication strength to each of them using the approach outlined in Sec. 5. The Intrinsic Authentication Strength for each PIV Authentication Use Case with its two components – Number showing theVerified Bindings and Negative Signs to denote bindings verified using weak authentication mechanisms – is shown in Table A.4 below:

Table A.4. SCIV-ALM Assigned Intrinsic Authentication Strength for PIV Authentication Use Cases

PIV Authentication Use Case	Intrinsic Authentication Strength
6.2.5 – CHUID -Authentication Using the CHUID	[1]
6.2.3.2 – PKI-CAK - Authentication with the Card Authentication Certificate	[3]
6.2.1.1 – BIO- Unattended Authentication Using PIV Biometric	[4 -] or [4 --]
6.2.1.2 – BIO-A- Attended Authentication of PIV Biometric	[4] or [4 -]
6.2.2 – OCC-AUTH - Authentication Using On-Card Biometric Comparison	[4] or [4 -]
6.2.3.1 – PKI-AUTH- Authentication with the PIV Authentication Certificate	[4] or [4 -]

A.4 SCIV-ALM Authentication Assurance Level Taxonomy for PIV Authentication Use Cases

Using the composition of verified bindings and intrinsic authentication strengths as metrics, we developed the partial order sequences in the total set of PIV Authentication Use Cases. These partial order sequences were then used to develop the SCIV-ALM Authentication Assurance Level Taxonomy for PIV Authentication Use Cases as shown in Fig. A.1 below. Each node in the taxonomy graph represents a PIV Authentication Use Case with its SCIV-ALM assigned intrinsic authentication strength shown within a square bracket.

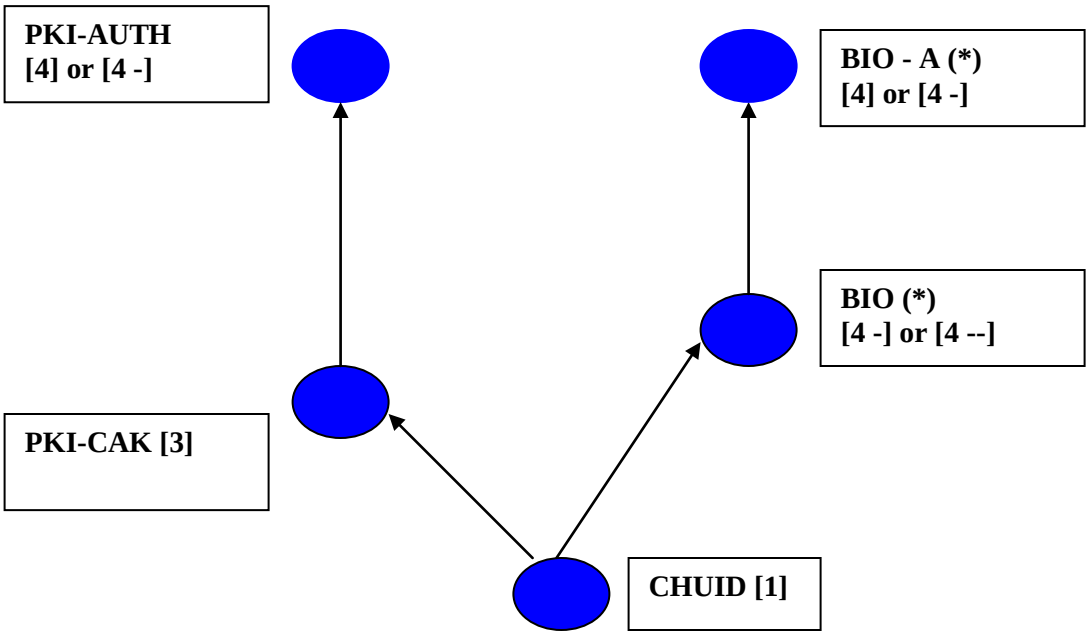

Figure A.1 SCIV-ALM Authentication Assurance Level Taxonomy for PIV Authentication Use Cases

(*) The Authentication Use Case OCC-AUTH may at the level of BIO –A or BIO depending upon whether the live sample is presented on the card in the presence of an attendant or not.

A.5 Comparison of Assigned Authentication Assurance Levels in PIV Specification and SCIV-ALM

In the following paragraphs we compare the authentication assurance levels assigned to PIV Authentication Use cases in the PIV Specification and by SCIV-ALM.

A.5.1 Hierarchical Authentication Assurance Levels between PKI-CAK and BIO

In the PIV specification, the authentication use case PKI-CAK is assigned the confidence level SOME while BIO is assigned the level HIGH, thereby implying that BIO clearly provides more authentication assurance than PKI-CAK. However on examining the set of verified bindings in these two PIV Authentication Use Cases, we find that PKI-CAK, by associating the unique person identifier with a cryptographic token secret provides the assurance that the person identifier is strongly bound to the particular copy of the token while the signed biometric object used in BIO provides the strong binding between the person identifier and the cardholder (through the signed biometric data). Thus we see that the set of bindings provided by PKI-CAK and BIO are incompatible. The incompatible assurance levels assigned to these PIV Authentication Use cases in our taxonomy diagram A.1 brings out this feature.

A.5.2 Identical Authentication Assurance Levels to BIO-A and PKI-AUTH

In the PIV specification, both the authentication use cases BIO-A and PKI-AUTH are assigned the same assurance level VERY HIGH. This is due to the fact that BIO-A provides strong resistance to use of unaltered card by non-owner (impersonation) using a combination of biometric and PIN while PKI-

AUTH, although provides the same assurance using just the PIN, compensates for this weakness by providing high resistance to credential forgery by associating the unique identifier with a cryptographic secret. However, examination of the verified bindings of each of these two authentication use cases brings out subtle differences in spite of the fact that each of these authentication use cases verify the same number of bindings (i.e., four). First we observe that both these authentication use cases share some common verified bindings. These are: Card Issuer – Person Identifier binding and Cardholder – Physical Token binding. Then if we consider the two remaining bindings in each of them, we find that BIO-A's bindings collectively perform authentication of the cardholder (due to combination of Person Identifier – Cardholder Trait and Cardholder Trait – Cardholer bindings) while PKI-AUTH's bindings collectively perform authentication of the physical token (due to combination of Person Identifier – Token Secret and Token Seccret – Physical Token bindings). Thus we see that apart from the common property of linking the cardholder to the physical token though the use of PIN, the authentication focus is different in these two authentication use cases. This is clearly brought out in our taxonomy diagram.

A.5.3 Identical Authentication Assurance Level to BIO-A and OCC-AUTH

The PIV specification assigns identical assurance level of VERY HIGH to both BIO-A and OCC-AUTH. However based on the observation that the key binding in both authentication use cases is the Cardholder Trait – Cardholder binding and that the strength of this binding depends upon the liveness property of the presented biometric sample and not the mechanism used for comparison, our taxonomy provides for the fact that OCC-AUTH can be either in BIO-A or BIO levels depending upon whether the presentation of the live sample directly to the card is performed under the supervision of an attendant or not.

Appendix B—Case Study: TWIC Authentication Use Cases

In this Appendix, we illustrate the application of SCIV-ALM methodology to Transportation Security Administration's identity verification program - Transporation Worker Identification Credential (TWIC). We present the case study of the application of SCIV-ALM to TWIC program's authentication use cases using the following steps:

- General overview of the TWIC program
- Brief description of TWIC Authentication Use Cases and their assigned assurance levels in TWIC's specification document [TWIC2008].
- The assignment of intrinsic authentication strengths to TWIC Authentication Use Cases using SCIV-ALM.
- The SCIV-ALM Authentication Assurance Taxonomy for TWIC Authentication Use Cases
- Comparison of authentication assurance levels assigned in the TWIC specification with the authentication assurance levels assigned by SCIV-ALM.

B.1 General Overview of the TWIC Program

The mission of the TWIC program is to design and field a common credential for all transportation workers requiring unescorted physical and logical access to secure areas of the nation's transporation system and their associated information systems [TWIC2008]. The TWIC credentials and the TWIC card application are carried in a smart card (henceforth referred to as TWIC card) whose physical token specification conforms to the PIV specification. In many cases, the TWIC card also contains the PIV card application.

B.2 Brief Description of TWIC Authentication Use Cases and Specified Assurance Levels

In TWIC specification, the successful outcome of an Authentication Use Cases results in designating the data object on the TWIC card that participates in the authentication protocol of that Use Case as an Acceptable Authentication (Assurance) factor. The consequence of this designation is that the unique identifier (i.e., FASC-N) it contains can be used as an input for the access control rules governing access to the resource that is protected by the TWIC card application. With this background information, the TWIC Authentication Use Cases and corresponding description are given in Table B.1 below.

Table B.1 – Brief Description of TWICAuthentication Use Cases

Authentication Mode	TWIC Authentication Use Case	Brief Description
A.1	CHUID Verification	Before using the CHUID object (containing the Unique Identifier FASC-N), its digital signature is verified (at least once) in order to ensure that it was issued by the right authority and has not been altered or tampered.
A.2	Active Card Authentication	A digital certificate called the Card

		Authentication Certificate is validated (to ensure that it was issued by a trusted CA) and proof of possession of the corresponding private key held by the card demonstrated through a Challenge-Response Protocol
A.3	CHUID Verification + Biometric User Authentication	The digital signature of the biometric template object (containing the Unique Identifier) is verified to ensure that it was issued by the authorized card issuer. The cardholder's live sample is matched against the stored biometric template
A.4	(CHUID) Signing Certificate + Active Card Authentication + Biometric User Authentication	Combines the process used in A.1, A.2 & A.3

In the TWIC specification, the assurance level for an Authentication Use case is based on the number and quality of each authentication factor. The quality (strength) of each authentication factor is determined by the level of difficulty for an attacker to gain control, clone or compromise that factor. An acceptable authentication factor is a data object that can be verified to be trusted enough such that the unique Identifier (i.e., FASC-N) it contains can be used as input to the Authorization system. Based on this logic, the designated authentication assurance levels for the TWIC Authentication Use Cases along with rationale are given in Table B.2 below.

Table B.2 - TWIC Authentication Assurance Levels & Rationale

Authentication Mode/TWIC Authentication Use Case (Authenticator Factor Data Object)	**Assigned Authentication Assurance Level**	**Rationale**
A.1 CHUID Verification (CHUID object digitally signed by the TWIC card issuer)	Verifiable Identification Factor (for our purpose can be looked upon as Weak Single Factor)	CHUID being accessible over the contactless interface can be captured, copied to another card or replayed along with the digital signature attached to it.
A.2 Active Card Authentication (A private cryptographic key + Certificate containing its public key counterpart that also holds the Unique Identifier & its Expiry Date)	Strong Single Factor Authentication	Provides proof of possession of a never revealed private key in the smart card chip that cannot be copied via any interface. The binding of the public counterpart of this key with the Unique Identifier in the Card Authentication Certificate issued by an authoritative Certificate Issuer

A.3 CHUID Verification + Biometric User Authentication (The biometric template object that includes the CHUID (unique identifier) and digitally signed by the TWIC card issuer)	Strong Single Factor Authenticaton	Successful matching of cardholder's live sample with the strored biometric establishes the strong binding between user's Unique Identifier (FASC-N) with the card holder
A.4 (CHUID) Signing Certificate + Active Card Authentication + Biometric User Authentication (Combination of artifacts used in A.1, A.2 & A.3)	Strong Two Factor Authentication	Using the authenticator factors in both A.2 & A.3 and matching the Unique Identifier (i.e., FASC-N) extracted from each of these factors

B.3 SCIV-ALM Assigned Intrinsic Authentication Strengths for TWIC Authentication Use Cases

We now assign intrinsic authentication strengths (in terms of the number and type of verified bindings) to TWICAuthentication Use Cases based on SCIV-ALM. In order to accomplish this, we examine the protocols involved in each Authentication Use Case and map them directly to the primitive authentication mechanisms of our methodology and their associated verified bindings (along with their types) as shown in Table B.3 below. This approach is possible because of the granularity of description of each TWIC Authentication Use Case in the TWIC specification document.

Table B.3. SCIV-ALM Verified Bindings for TWIC Authentication Use Cases

Authentication Mode/TWIC Authentication Use Case (Authenticator Factor Data Object)	**SCIV-ALM Primitive Authentication Mechanisms involved**	**SCIV-ALM Verified Bindings (Type)**
A.1 CHUID Verification (CHUID object digitally signed by the TWIC card issuer)	**PUM-1**: Person Identifier's origin, status, and integrity checked using its associated digital signature	*Card Issuer – Person Identifier Binding (Strong)*
A.2 Active Card Authentication (A private cryptographic key + Certificate containing its public key counterpart that also holds the Unique Identifier & its Expiry Date)	**PUM-1** functionality provided through equivalent operation **PUM-2.1**: Establishing Trust in the Card Authentication Certificate (ensuring that the certificate was issued by a trusted, authorized CA, is currently active and that the digital signature generated by the certificate issuer verifies), The unique person identifier is obtained from the certificate **PUM-3**: Verifying Presence of non-shared embedded token secret (tested by sending an input data from the Verifier and verifying the token response through a related	*1.* (*Card Issuer – Person Identifier Binding (Strong)) (Implicit)* *2. (Person Identifier – Token Secret Binding (Strong))* *3. (Token Secret – Physical Token Binding (Strong))*

	artifact) (e.g., checking the presence of asymmetric private key by verifying the signed response using its associated validated public key)	
A.3 CHUID Verification + Biometric User Authentication (The biometric template object that includes the CHUID (unique identifier) and digitally signed by the TWIC card issuer)	**PUM-1**: Person Identifier's origin, status, and integrity checked using its associated digital signature **PUM-6**: The signed biometric object is read after performing the necessary operations. Trust in the signing certificate established through Certificate Validation and the digital signature in the signed biometric object verified **PUM-7.1**: The cardholder presents a live sample of biometric data in an unattended authentication station (We are assuming this since TWIC does not specify this aspect)	*1. Card Issuer – Person Identifier Binding (Strong)* *2. Person Identifier – Cardholder Trait Binding (Strong)* *3. Cardholder Trait – Cardholder Binding (Weak)*
A.4 (CHUID) Signing Certificate + Active Card Authentication + Biometric User Authentication (Combination of artifacts used in A.1, A.2 & A.3)	**PUM-1** + **PUM-2.1** + **PUM-3** + **PUM-6** + **PUM-7.1**	*1. Card Issuer – Person Identifier Binding (Strong)* *2. (Person Identifier – Token Secret Binding (Strong))* *3. (Token Secret – Physical Token Binding (Strong))* *4. Person Identifier – Cardholder Trait Binding (Strong)* *5. Cardholder Trait – Cardholder Binding (Weak)*

Using the number and type of verified bindings for each PIV Authentication Use Case, we assign the intrinsic authentication strength to each of them using the approach outlined in Sec. 5. The Intrinsic Authentication Strength for each PIV Authentication Use Case with its two components – Number showing the Verified Bindings and Negative Signs to denote bindings verified using weak authentication mechanisms – is shown in Table B.4 below:

Table B.4. SCIV-ALM Assigned Intrinsic Authentication Strength for TWIC Authentication Use Cases

Authentication Mode/TWIC Authentication Use Case	Intrinsic Authentication Strength
A.1 CHUID Verification	[1]
A.2 Active Card Authentication	[3]
A.3 CHUID Verification + Biometric User Authentication	[3 -]
A.4 (CHUID) Signing Certificate + Active Card Authentication + Biometric User Authentication (Combination of artifacts used in A.1, A.2 & A.3)	[5 -]

B.4 SCIV-ALM Authentication Assurance Level Taxonomy for TWIC Authentication Use Cases

Using the composition of verified bindings and intrinsic authentication strengths as metrics, we developed partial order sequences in the total set of TWIC Authentication Use Cases. These partial order sequences were then used to develop the SCIV-ALM Authentication Assurance Level Taxonomy for TWICAuthentication Use Cases as shown in Fig. B.1 below. Each node in the taxonomy graph represents a TWIC Authentication Use Case with its SCIV-ALM assigned intrinsic authentication strength shown within a square bracket.

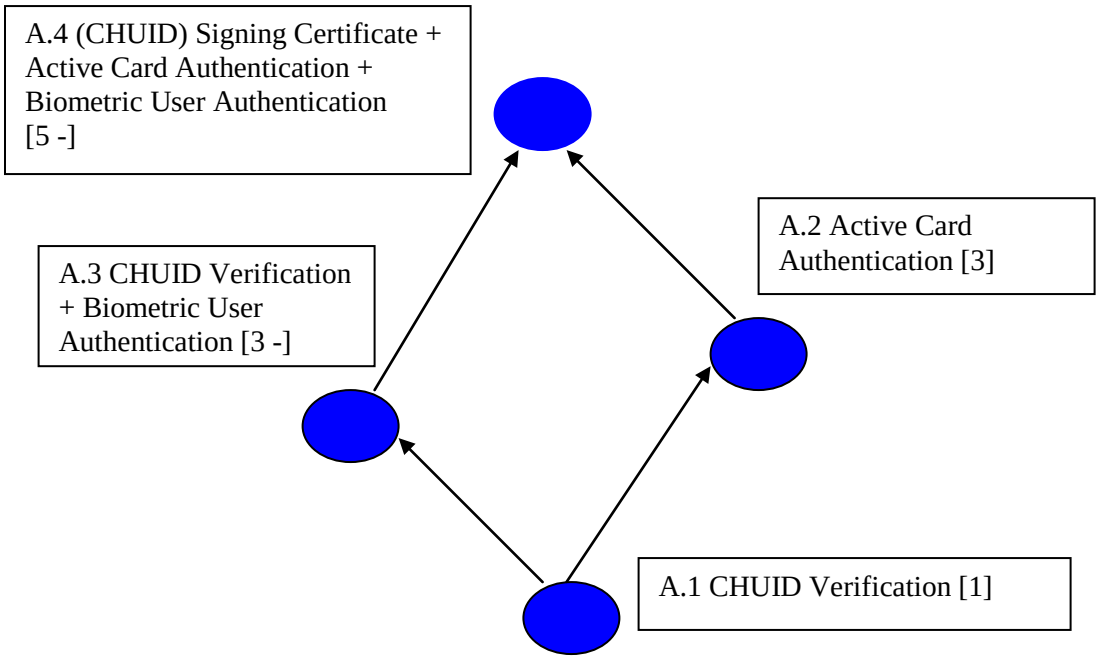

Figure B.1 SCIV-ALM Assurance Level Taxonomy for TWIC Authentication Use Cases

B.5 Comparison of Assigned Authentication Levels in TWIC Specification and SCIV-ALM

In the following paragraphs we compare the authentication assurance levels assigned to TWIC Authentication Use cases in the TWIC Specification and by SCIV-ALM.

B.5.1 Direct Traceability to Trust Link established during Card Issuance in SCIV-ALM

The designation of authentication assurance level for any TWIC Authentication Use Case does not reflect the trust link that data object (used in that use case) embodies. For example, according to the TWIC specification, the Card Authentication Key and the Certificate are to provide "a mechanism that strongly binds the cardholder's identity (via the FASC-N) to the physical card token by embedding a piece of secret data in the chip that cannot be copied via any interface. This key data may be used in conjunction with the freely readable certificate to prove that the card has not been cloned or spoofed" [TWIC2008]. However, the assignment of "Single Factor Authentication" designation to Active Card Authentication Use Case (Authentication Mode A.2) does not reflect the fact that the Card Authentication Key & Certificate (the objects involved) establishes the strong binding between the cardholder's identity and the physical token. The SCIV-ALM methodology, by directly associating the "Person Identifier – Token Secret Binding (Strong))" and "Token Secret – Physical Token Binding (Strong))" as verified bindings directly reflects the embodied trust in the objects used in Authentication Mode A.2.

B.5.2 Providing Distinguishing Criteria for choosing between two Use Cases at the same Assurance Level in SCIV-ALM

Another feature of SCIV-ALM is the distinguishing criteria it provides for choosing one Authentication Use Case over another among Use Cases at the same Assurance Level for a particular context. For example, for an access control situation which does not require very high authentication assurance, both Active Card Authentication (Authentication Mode A.2) and CHUID Verification + Biometric User Authentication (Authentication Mode A.3), being at the same assurance level (i.e, Single Authentication factor) are equally eligible candidates. Hence, justification for choosing one Authentication Use Case over another cannot be provided using the TWIC Assurance Level designation alone. However, an Assurance Level designation that reflects the properties satisfied by each of these Authentication Use Cases not only will provide the justification for choosing one over another but also can directly associate the choice with the requirements of the access control context. In our methodology the set of verified bindings associated with Biometric User Authentication (A.3) and Active Card Authentiation (A.2) directly reflects the fact that these Authentication Use Cases satisfy non-intersecting (different) properties. Further, in the context of an Access Control application, where the requirement for associating a valid identifier with the right credential holder is more critical than the association of the valid identifier with an authorized physical token, the Authentication Use Case that must be chosen should be the Biometric User Authentication rather than Active Card Authencation, even though both are of the same Assurance level by the TWIC specification.