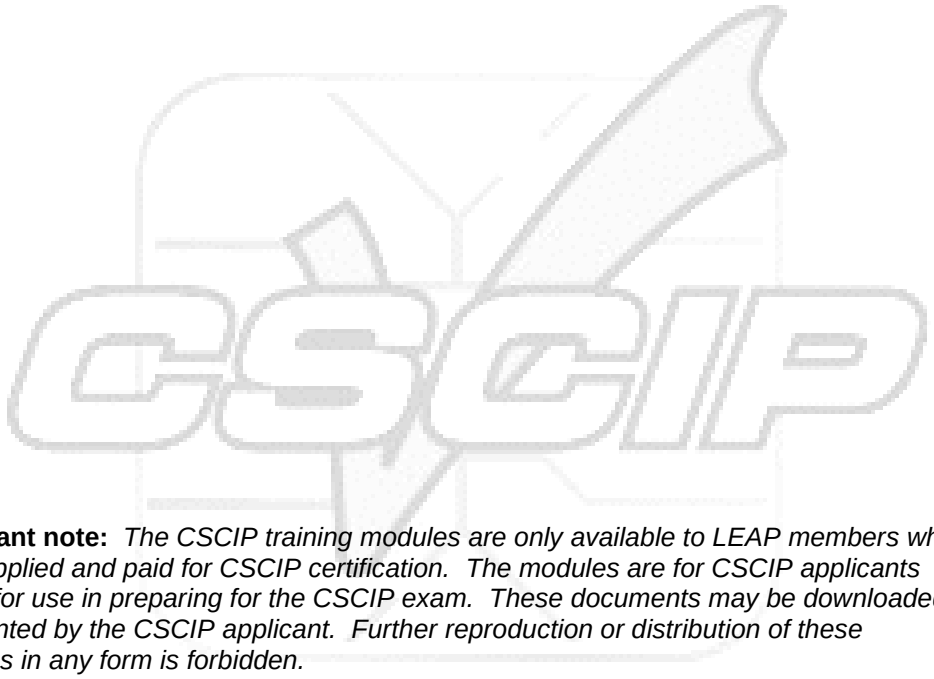
Module 2: Security

Smart Card Alliance
Certified Smart Card Industry Professional
Accreditation Program

About the Smart Card Alliance

The Smart Card Alliance is a not-for-profit, multi-industry association working to stimulate the understanding, adoption, use and widespread application of smart card technology. Through specific projects such as education programs, market research, advocacy, industry relations and open forums, the Alliance keeps its members connected to industry leaders and innovative thought. The Alliance is the single industry voice for smart cards, leading industry discussion on the impact and value of smart cards in the U.S. and Latin America. For more information, please visit http://www.smartcardalliance.org.



Important note: *The CSCIP training modules are only available to LEAP members who have applied and paid for CSCIP certification. The modules are for CSCIP applicants ONLY for use in preparing for the CSCIP exam. These documents may be downloaded and printed by the CSCIP applicant. Further reproduction or distribution of these modules in any form is forbidden.*

TABLE OF CONTENTS

1 Introduction

Security is a core element of any payment and identity system; a properly-designed system is not dependent on the security of any single component. No single security mechanism provides complete security and, indeed, complete security does not exist. The objective in any secure system design must be to implement the appropriate security measures to address the expected risks and threats to the system.

Smart card technology is a critical element in most secure payment and identity system designs worldwide, enabling organizations to provide citizens, consumers and employees with a secure, portable device that protects personal information and enables secure, authenticated transactions.

Smart card technology provides security benefits at a number of levels. The secure microcontrollers used in smart cards have both hardware and software security features manufactured into the ICs that thwart attackers from accessing any sensitive information that is stored in the card. The secure microcontroller also enables the smart card to interact intelligently with the reader and the system, implementing cryptographic functions that authenticate the card and cardholder to the system and the reader and system to the card. With contact and secure contactless interfaces, increasingly powerful processors, wide range of memory options, and flexible implementation of both symmetric and asymmetric cryptographic algorithms, smart card technology is a critical component in the chain of trust in a secure system design. Organizations implementing smart card-based systems can also look to a number of industries for best practices in system design and for resources for evaluating and certifying the security of smart card products.

This module describes the fundamentals of smart card and smart card-based system security. After reviewing this module, CSCIP applicants should be able to answer the following questions:

- What industry certifications and evaluations are available that organizations can use to gain confidence in the security implemented in various smart card products and in the interoperability of the technology among various component suppliers?
- What security features are designed into secure memory ICs and secure microcontrollers that protect data and thwart attempted attacks?
- What is the impact of contact and contactless interfaces on security?
- How is security maintained at the card edge interface?
- What are common cryptographic methods and algorithms? How do smart cards use cryptography to protect and improve the security of the overall system being implemented?
- What is PKI and how is it deployed?
- How do smart cards fit into overall system security? How are different industries using smart cards to improve the security of their applications?

2 Industry Certifications and Evaluations[1]

The government and financial payments industries have led the way in establishing security evaluation and certification programs for the various layers of smart card security. This section describes the industry-specific security evaluations for secure ICs, operating systems, and application software, as well as the entities that either require or perform these evaluations.

These standardized evaluations and certifications use a very few trusted third party labs to empirically verify that specific threats (that are state-of-the-art at the time) are prevented to a defined level of effectiveness. Measures of effectiveness encompassed in these standards include expertise, time, and cost of equipment required to achieve the specific attack. Equally important to the verification function, such standardized evaluations and certifications provide a framework to publish results of testing without disclosing details of the countermeasures that are used and verified. The resulting confidentiality allows smart cards to have their most effective security countermeasures tested without attackers knowing specifically what these countermeasures are. Best of all, those applying or specifying smart cards need not consider the specific hardware countermeasures (such as those described in this paper), but need only require that their card meet the required level of certification.

Smart cards are also subject to rigorous functional and interoperability testing, which is outside the scope of this white paper.

2.1 Proprietary versus ISO Standardized Security Evaluations/Certifications

Certification/evaluation schemes for smart cards can use an industry standardized and layered approach which is stepwise applied to the IC, then operating system (or fixed mask), and then application/applet; or a proprietary scheme which must be verified on each end product to be deployed. Openly created and published methodologies facilitate the security industry working together to define how each piece of a smart card value chain works together to deliver a complete and secure solution withstanding defined threats. Each piece of the value chain can reuse the prior step's certification/evaluation to achieve their own. For example, the IC supplier has had their secure microcontroller IC evaluated and it has received a Common Criteria (CC) EAL5+ certification. The IC manufacturer prepares a specification instructing their customers on how to make use of the IC's security features to code an operating system (or fixed function mask) that is also CC certifiable. The card operating system vendor submits the finished product to a verification lab for CC EAL4+ certification. Upon receiving a certification for their OS/mask, the supplier provides a specification guiding customers to apply the product as necessary to achieve a certifiable application of the finished card.

The differences in cost and change management for each of these approaches are both obvious and great. The standardized approach allows reuse of the prior step's work with verifiers checking to ensure the OS provider (or application user) has correctly used the features that have already been verified. A change required to the applet would only require the applet on the verified IC and OS to be rechecked using the standardized approach; this would require a full resubmission using the proprietary approach. As expected, the costs of rechecking the entire platform are significant and being able to reuse parts of the verification process greatly reduces the cost and time to complete the evaluation/certification process.

2.2 ISO/IEC 15408 – Common Criteria

Common Criteria (CC) is an internationally approved security evaluation framework providing a clear and reliable evaluation of the security capabilities of IT products, including secure ICs, smart card operating systems, and application software. CC provides an independent assessment of a product's ability to meet security standards, with the goal of giving customers confidence in the security of IT products and leading to better decisions about security. Security-conscious customers, such as national governments, are increasingly requiring CC certification in making purchasing decisions. Since the requirements for

[1] *What Makes a Smart Card Secure?*, Smart Card Alliance white paper, October 2008.

certification are clearly established, vendors can target very specific security needs while providing broad product offerings.

CC has been adopted and is recognized by 26 countries, which allows customers in any of these countries to purchase products with the same level of confidence. Evaluating a product with respect to security requires identification of the customer's security needs and an assessment of the capabilities of the product. CC helps customers complete both of these processes using two key tools: protection profiles and evaluation assurance levels.

2.2.1 Protection Profiles

A protection profile defines a standard set of security requirements for a specific type of product. Protection profiles are the basis for the CC evaluation. By listing required security features for specific product families, CC enables products to achieve conformity to a relevant protection profile. During CC evaluation, each product is tested against a specific protection profile, providing reliable verification of the security capabilities of the product.

For smart cards, the protection profile covers secure ICs, smart card operating systems, and application software. These components can be evaluated as separate entities or combined into a secure smart card. More than 53 protection profiles for secure ICs, smart card operating systems, application software, and other smart card related devices and systems are listed on the CC portal.[2] Customers can compile a list of critical security features by examining the details of the relevant protection profiles. The CC certification verifies that a product meets the requirements of that protection profile. Using a CC certification, customers can rapidly assess a product's ability to meet their security needs and compare the security capabilities of different validated products.

Two popular protection profiles are:

- The E-passport Protection Profile, version 2.1[3]
- The Security IC Platform Protection Profile Version 2.0, known as BSI-CC-PP-0084-2014. This protection profile is established by Eurosmart and the smart card IC industry and is an update of the Smartcard IC Platform Protection Profile CC -PP_0035-2007. Both contact and contactless smart cards use this protection profile.

\

Multi-application smart card operating systems also use CC for security evaluations. For example, the Java Card Protection Profile is available as a collection of four separate protection profiles. One profile defines a set of security requirements for the Java Card Runtime Environment, another the Java Card Virtual Machine, a third the Java Card API Framework, and the fourth the on-card installer components. The profile provides guidelines for developing a secure Java Card platform and defines a security target that must be met to obtain high-level security certifications. The Java Card Protection Profile is intended to complement other protection profiles currently available for smart cards based on Java Card technology.[4]

In the past, MULTOS implementations have been evaluated against the Smart Card Security User Group (SCSUG) Smart Card Protection Profile v2.0.[5]

Proprietary operating systems are usually evaluated against a specific protection profile in conjunction with a specific application. For example, the CC portal lists protection profiles for healthcare cards, ePassports, contactless cards, and electronic purses.

Additional Common Criteria profiles for smart cards can be found at http://www.commoncriteriaportal.org/pps/IC/#IC.

[2] http://www.commoncriteriaportal.org/pp_IC.html#IC
[3] http://www.commoncriteriaportal.org/files/ppfiles/20110221143918.pdf
[4] http://java.sun.com/javacard/pp.html
[5] http://www.multos.com/downloads/marketing/Whitepaper_MULTOS_Security.pdf

2.2.2 Evaluation Assurance Levels

An evaluation assurance level (EAL) measures the depth of engineering review and evaluation of the product lifecycle. Unlike a protection profile, the EAL does not indicate the actual security capabilities of the product but independently stipulates the level of evidence reviewed and tested against the vendor's claims. Figure 1 shows the seven CC EALs (EAL1–EAL7) and the level of testing required to achieve the different levels. Vendors can choose an EAL.

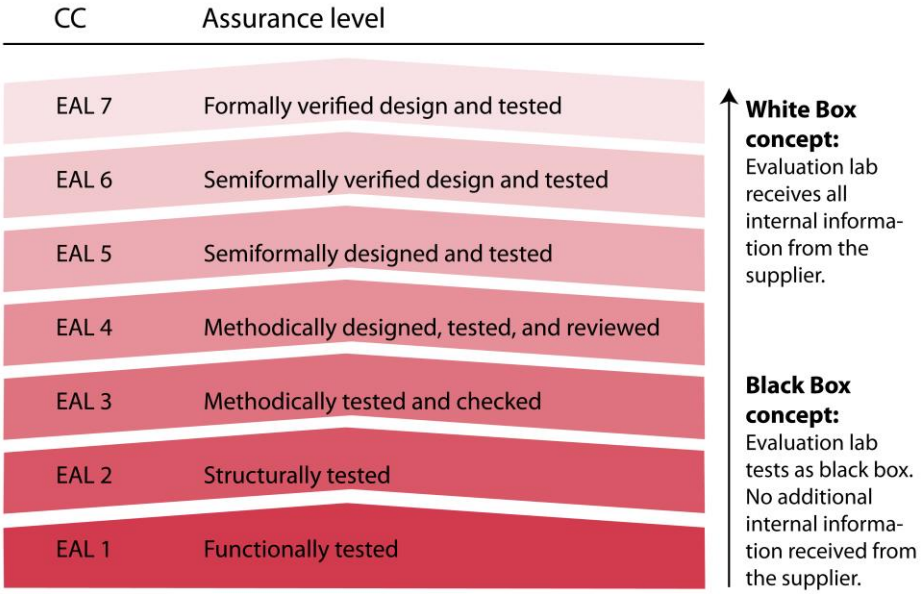


Figure 1. Common Criteria Evaluation Assurance Levels

2.2.3 Certification Process

The purpose of product certification is to provide customers with a high level of trust, which requires a thorough, reliable, objective, and globally accepted process. To submit a product for certification, the vendor must first specify a security target (ST). The ST description includes an overview of the product, potential security threats, detailed information on the implementation of all security features included in the product, and any claims of conformity against a protection profile at a specified EAL. The vendor must submit the ST to an accredited testing laboratory for evaluation. The laboratory then tests the product to verify the described security features and evaluate the product against the specified protection profile. A successful evaluation results in official certification of the product against a specific protection profile at a specific EAL. Most smart card components are currently certified to EAL 4+ and are moving to EAL 5+, as indicated in the CC portal.[6]

2.3 FIPS 140 for Cryptographic Modules

FIPS 140 is the U.S. and Canadian government security standard for cryptographic modules. It applies to the entire smart card, including the secure IC, the operating system, and the application software. This standard is the benchmark for implementing cryptographic software and hardware and specifies best practices for implementing cryptographic algorithms, handling key material and data buffers, and securely working with the operating system. In the late 1990s, smart card manufacturers began submitting smart cards for FIPS 140-1 certification. In 2001, FIPS 140-1 was replaced by FIPS 140-2, and FIPS 140-3 will eventually replace FIPS 140-2.

[6] For an overview of the portal, see http://www.commoncriteria.org/introductory_overviews/CCIntroduction.pdf. For an introduction and general model, see http://www.commoncriteria.org/docs/pdf/CCPART1V21.PDF

FIPS 140 specifies the requirements for cryptographic modules in the areas of secure design and implementation, including module specification, ports and interfaces, roles, services, and authentication, finite state model, physical security, operational environment, cryptographic key management, electromagnetic interference/electromagnetic compatibility (EMI/EMC), self-tests, design assurance, and mitigation of other attacks.

FIPS 140-2 specifies four levels of security. The standard does not specify what level is required by any particular application.

- Level 1 imposes very limited requirements; all components must be "production-grade" and obvious security functions must be present. Level 1 restricts the machine on which the module runs to operating in single-user mode. Level 1 can apply to software-only implementation.
- Level 2 adds requirements for physical tamper-evidence and role-based authentication. It is noticeably harder to obtain. The difficulty is not necessarily with the cryptographic module code, but rather with the formalities required and the fact that Level 2 modules must run on validated hardware under validated operating systems.
- Level 3 adds requirements for physical tamper-resistance and identity-based authentication. Level 3 also requires physical or logical separation between the interfaces by which certain security parameters enter and leave the module.
- Level 4 imposes much more onerous physical security requirements and requires more robust security features to defend against various environmental attacks.

Cryptographic modules receive security level ratings that reflect the requirements they meet. Most smart cards (secure IC plus OS plus application software) that are certified by FIPS 140 are certified to either Level 2 or Level 3. These certifications are granted according to the Cryptographic Module Validation Program (CMVP), a joint American and Canadian security accreditation program for evaluating and certifying cryptographic modules. All of the tests under the CMVP are handled by third-party laboratories that are accredited as Cryptographic Module Testing Laboratories by the National Voluntary Laboratory Accreditation Program (NVLAP).

FIPS 140-3 is the proposed revision to FIPS 140-2. The draft specifies five security levels instead of four, provides a separate section for software security, requires mitigation of non-invasive attacks when validating at higher security levels, introduces the concept of public security parameters, allows certain self-tests to be deferred until specific conditions are met, and strengthens the requirements for user authentication and integrity testing. The additional security level specified by FIPS 140-3 incorporates extended and new security features that reflect recent advances in technology.[7]

2.3.1 FIPS 140 Compared to Common Criteria

The FIPS 140 standard was initially developed in 1994, before the development of CC and with a different goal. FIPS 140 evaluates a defined cryptographic module and provides a suite of conformance tests with up to four security levels to determine whether the module meets certain security requirements. FIPS 140 prescribes basic requirements for cryptographic modules, including requirements in areas such as physical security, key management, self tests, and roles and services. A FIPS140 certification/evaluation applies only to the finished cryptographic module and does not allow a composite approach on the various pieces comprising the module (such as hardware/IC, operating system, and applet).

CC evaluates a security target against an industry-defined protection profile. A protection profile typically applies to a broad range of products, while FIPS 140 certification applies to only a single product. A CC evaluation does not supersede or replace FIPS 140 validation. The four security levels in FIPS 140-2 do not map directly to specific CC EALs or to CC functional requirements. A CC certificate cannot be substituted for a FIPS 140-1, FIPS 140-2, or FIPS 140-3 certificate. CC has been designed/planned to use composite evaluations, as described in Section 2.2.

[7] For more information on FIPS 140-2, FIPS 140-3, and the CMVP, see http://csrc.nist.gov/groups/STM/index.html.

2.4 Industry Evaluations of Security and Applications

Different industries have additional requirements for testing and certifying smart card products, at either the security or application level or both. Two examples are discussed in this section – FIPS 201 evaluation that is use by the U.S. Federal government and the payments industry's evaluation.

2.4.1 FIPS 201 for Application Software

Homeland Security Presidential Directive 12 (HSPD-12), issued on August 27, 2004, mandated the establishment of a standard for identification of Federal government employees and contractors. HSPD-12 requires the use of a common identification credential for both logical and physical access to federally controlled facilities and information systems.

The Department of Commerce and the National Institute of Standards and Technology (NIST) were tasked with producing a standard for secure and reliable forms of identification. In response, NIST published Federal Information Processing Standard Publication 201 (FIPS 201), *Personal Identity Verification (PIV) of Federal Employees and Contractors*, initially issued on February 25, 2005 and updated to FIPS 201-1 in March 2006. The FIPS 201 PIV card is a smart card with both contact and contactless interfaces that is to be used for both physical and logical access control and other applications as determined by the individual agencies.

FIPS 201 consists of two parts: PIV I and PIV II. (Note that PIV-I stands for PIV Interoperable; PIV I refers to PIV policies and PIV II for technical standards.) The standards in PIV I support the objectives for identity assurance and requirements for determining trustworthiness described in HSPD-12. The standards in PIV II support the technical interoperability requirements described in HSPD-12. PIV II also specifies standards for implementing identity credentials on IC cards (i.e., smart cards) for use in a Federal PIV system.

All cryptographic functions on a FIPS 201 PIV card must be evaluated and certified under the FIPS 140 specifications.

In addition, NIST has established the NIST Personal Identity Verification Program (NPIVP)[8] to validate the PIV components required by FIPS 201. The two objectives of the NPIVP program are

- To validate the compliance/conformance of two PIV components, PIV middleware and PIV card application, with the specifications in NIST SP 800-73-4, parts 1-3, May 19, 2014, *Interfaces for Personal Identity Verification*.
- To provide assurance that the set of PIV middleware and PIV card applications that have been validated by NPIVP are interoperable

The General Services Administration (GSA) has also established the FIPS 201 Evaluation Program[9] to evaluate products and services offered for use in HSPD-12 and to ensure that products and services are compliant with established FIPS 201 requirements. Products or services that are evaluated and comply with FIPS 201 specifications are added to the GSA Approved FIPS 201 Products and Services List.[10]

2.4.2 Financial Payment Industry: EMVCo Security Evaluations

In the past, individual payment brands had been solely responsible for defining security requirements and establishing evaluation procedures for secure ICs and for smart cards (also known as integrated circuit cards (ICCs)). Beginning in 2007, the EMVCo[11] Security Evaluation Working Group (SEWG) assumed

[8] http://csrc.nist.gov/groups/SNS/piv/npivp/index.html
[9] http://fips201ep.cio.gov/
[10] http://fips201ep.cio.gov/apl.php
[11] EMVCo LLC was formed in February 1999 by Europay International, MasterCard International, and Visa International to manage, maintain, and enhance the EMV Integrated Circuit Card Specifications for Payment Systems. With the acquisition of Europay by MasterCard in 2002 and with JCB joining the organization in 2004, EMVCo is currently operated by JCB International, MasterCard Worldwide, and Visa, Inc.

responsibility for evaluating all EMV-based contact smart card ICs. In 2008, this responsibility was extended to contactless smart card ICs as well. In addition, the SEWG is responsible for evaluating the security of any implementations of the EMVCo Common Payment Application (CPA). The individual payment brands (American Express, Discover, JCB, MasterCard, Visa) still maintain responsibility for evaluating the security of their individual payment applications, regardless of whether these applications are contact or contactless.

The primary objective of the EMVCo security evaluation process is to ensure that ICs and CPA smart cards conform to EMVCo security guidelines. The IC security evaluation includes the firmware and software routines required to access the security functions of the IC. The CPA smart card security evaluation includes the IC, the operating system, and all common payment applications that reside on the smart card.

The EMVCo Security Evaluation Secretariat is responsible for administering the EMVCo security evaluation process. By common agreement, MasterCard Worldwide performs the functions of the EMVCo Security Evaluation Secretariat, using the resources of the MasterCard Analysis Laboratory (MCAL).

The methodology used in the evaluation process leverages a program of research targeted at attack methodology. In addition, EMVCo supports the work of the International Security Certification Initiative (ISCI) and will support ongoing security initiatives under proposed Joint Interpretation Library (JIL) leadership, to maintain a common set of threats and attacks.

2.4.2.1 IC Security Evaluation

The IC security evaluation considers the security of the IC product used in the smart card and is intended to provide a high level of confidence in the security functions that are designed to deal with known attack methods. The EMVCo security evaluation process also takes into account the security of the design, development, and delivery processes. The IC security evaluation is performed by recognized external security evaluation laboratories and funded by the product provider.

2.4.2.2 CPA Smart Card Security Evaluation

The CPA smart card security evaluation considers the security of the product providers who develop operating systems and payment applications and evaluates how these applications and operating systems follow the relevant security guidelines. An important factor is how the product providers build upon the security of the IC and the OS to provide overall security for a payment application on the smart card.

2.4.2.3 Certification Process

The EMVCo security evaluation process has been conceived to provide a “high” level of assurance, as defined in the document *Application of Attack Potential to Smartcards*[12] for IC and CPA smart card products at all stages of the development process. At the device level, the evaluation methodology tries to balance so-called “black box” and “white box” testing. This balance is achieved by carrying out a security analysis that considers all viable attacks on a product and derives a set of penetration tests based on individual device characteristics.

Recognized external evaluation laboratories perform security evaluations using the relevant EMV Security Guidelines and externally developed testing tools. EMVCo recognizes the methodology used by some formal evaluation schemes (e.g., Common Criteria) but will only accept full evaluation reports as evidence.

The output from the EMVCo security evaluation process is an EMVCo Compliance Certificate that includes:

[12] Joint Interpretation Library, "Application of Attack Potential to Smartcard," version 2.5, November 2007, http://www.ssi.gouv.fr/site_documents/JIL/JIL-Application-of-Attack-Potential-to-Smartcards-V2-5.pdf

- A number that identifies a single approval path from product provider through manufacturer to issuer
- A date that reflects the status of the EMVCo security guidelines at the time of evaluation

Product providers must present their EMVCo Compliance Certificate number to issuers as proof that their product has been evaluated by the EMVCo security evaluation process.[13]

2.4.3 Financial Payment Industry: Other Product Security Evaluations

In addition to EMVCo security evaluations for EMV credit and debit cards, the different international payment brands (American Express, Discover, JCB, MasterCard and Visa) have specific security evaluations for their unique payment applications for both magnetic stripe cards and smart cards.

In addition to these international payment brand credit and debit cards, certain countries have proprietary debit and stored value payment cards. Typically, the issuers of these cards will require a CC security evaluation prior to card issuance.

2.5 Operating System Evaluation

In addition to IC and application level evaluation and certification, card operating systems are also evaluated.

Most of the major card manufacturers have developed proprietary operating systems that have been certified to Common Criteria and/or FIPS 140. In addition to proprietary operating systems, both Java Card and MULTOS offer security evaluation resources.

The Java Card Protection Profile[14] is available as a collection of four protection profiles. A profile defines a set of security requirements for the Java Card runtime environment, the Java Card virtual machine, the Java Card API framework, and the on-card installer components. The profile provides guidelines to develop a secure Java Card platform and define a security target in order to obtain high-level security certifications. The Java Card Protection Profile is intended to complement existing protection profiles available for Java Card technology-based smart cards.

MULTOS has a standard security assurance target which all implementations of the MULTOS operating system must be evaluated against.[15] Specific implementations have been given an ITSEC E6 accreditation (EAL 7 Common Criteria) by the UK and Australian Governments. MAOSCO, the managing secretariat of the MULTOS Consortium, administers the Type Approval policy of the MULTOS scheme, of which the platform security evaluation is part. An implementer can only sell a product that is has been developed, evaluated and certified in accordance with the process described in the MULTOS Type Approval Policy document.

2.6 Comparison of Security Evaluations

Table 1 summarizes the types of security evaluations described in this document and corresponding certifications.

[13] A list of approved ICs can be found at http://www.emvco.com/securityevaluation.asp?show=97.
[14] http://java.sun.com/javacard/pp.html
[15] http://www.multos.com/downloads/marketing/Whitepaper_MULTOS_Security.pdf

Table 1. Currently Available Security Evaluations and Certifications

	Integrated Circuit Evaluation	Operating System Evaluation	Operating System and Application Evaluation	Notes
FIPS 140	✓	✓	✓	U.S. government standards; applies to cryptographic module only
FIPS 201			✓	U.S government standard
ISO/IEC 15408 Common Criteria	✓	✓	✓	Cross industry
EMVCo	✓		✓	Payments industry
Payment Brands			✓	Payments industry

3 Integrated Circuit Level Security

Although a smart card IC can be either a secure memory IC or a secure microcontroller, this module focuses on the secure microcontroller. Secure microcontrollers support the confidentiality, authentication, and integrity of stored data while still allowing the data to be accessible for applications. Applications requiring the most security use microcontroller-based smart cards; examples of deployed applications include the ePassport, the FIPS 201 Personal Identity Verification (PIV) card, EMV[16] credit/debit cards, and contactless credit/debit cards.

3.1 Goals for IC Security

Acceptable security exists when the cost of a successful attack is an order of magnitude higher than the potential profit. Achieving security is an ongoing race. Given enough time, effort, and money, any security solution can be compromised.

The level of security implemented must be balanced appropriately for the data any given transaction uses. The value of the data being protected determines the level of security measures that should be deployed and the robustness of the cryptography that should be used. As the number of applications that are developed for secure ICs grow, the more attackers' attention is focused on the technology.

3.1.1 Types of Attackers

Attackers typically fall into one of three areas:

- Amateur. Amateurs are curious individuals who carry out attacks just to “see if it can be done.”
- Expert. Experts attack under the auspices of scientific institutions and universities studying the technology.
- Professional. Professionals attack for financial reward or to obtain sensitive data and compromise a system.

3.1.2 Types of Attacks

Attacks are techniques implemented to compromise the security of a smart card IC by discovering what information it holds. Attacks can be categorized as fault attacks, side-channel attacks, or invasive attacks.

Fault attacks alter the IC's internal workings to induce an error in the operation of the IC. Erroneous operation reveals information about the chip. The IC has a set of sensors that control IC operation (described in Section 3.2.1), as well as redundant logical operations. If the IC is manipulated to function outside the established sensor parameters, the IC goes into alarm mode or prevents operation completely.

Side-channel attacks are attacks based on information gained from the physical implementation of a cryptosystem. For example, timing information, power consumption, electromagnetic leaks, or even sound can provide a source of information that can be exploited to break the system. Many side-channel attacks require considerable technical knowledge of the internal operation of the system on which the cryptography is implemented.

Certain countermeasures in an IC can deter side-channel attacks (see Section 3.2.1):

- Random wait state insertion
- Bus confusion and memory encryption

[16] Europay MasterCard Visa. Specifications developed by Europay, MasterCard and Visa that define a set of requirements to ensure interoperability between payment smart cards and terminals.

- Continuous check of random characteristics
- Current scrambling/stabilizing
- Voltage regulation
- Dual bus rails, where the transmission of data is passed from one rail of the bus to the other to confuse the attacker

Invasive attacks, also known as **hardware attacks**, use more intrusive means to access the information on the IC. Examples of invasive attacks are probing the IC with a microprobe or focused ion beam (FIB), reverse engineering, and circuit modification.

Certain countermeasures implemented in an IC can deter invasive attacks (see Section 3.2.1):

- Flexible and user-defined memory encryption of user memory, RAM, and ROM
- Use of a memory management unit to prohibit one application from accessing the code of another application
- Active shielding that renders the IC inactive when triggered
- Small IC geometry (0.22 µm as a maximum feature size) to deter microprobing
- Bus confusion and encryption of data travelling on the bus
- Continuous checking of the random characteristics of the IC
- Proprietary timing and IC layout.

3.2 Achieving IC Security

The most comprehensive IC security is multi-dimensional. No single security mechanism protects completely against the broad spectrum of possible attacks. Therefore, the design of a secure IC and its use in a system must incorporate hardware, software, and system countermeasures to protect data and transactions.

Security should be an integral part of every smart card solution deployed. It is important to consider the security strength of the IC platform selected for any smart card application. Overall system security would also be enhanced by other measures implemented at the system level.

Secure smart card microcontrollers are commercially available that are designed to function in hostile environments. These ICs are fortified with mechanisms that are designed to withstand attempts to extract the confidential data the IC is protecting.

3.2.1 Secure Microcontroller Architecture

To defend against attacks, a secure IC should have an architecture that allows the IC to withstand all known attack types. Each IC manufacturer incorporates its own features and security modules into its IC architecture. The manufacturer may utilize its own nomenclature for the modules, but the modules perform similarly or identically while providing varying levels of protection. This section describes security features generically, recognizing that each manufacturer may have different terminology and varying levels of protection.

As described in Section 2, independent third party test laboratories can verify that each specific secure IC platform adequately protects itself from known/defined threats. Many IC manufacturers use feedback from these third party labs to improve and invent new countermeasures that they would never willingly share with their competition. Therefore, it is better to specify which threats the IC must be capable of resisting (and to what degree) than to specify the countermeasure, as described in the section below. Specifying the countermeasures might needlessly restrict selection of ICs or add cost while providing no benefit.

Figure 2 is a block diagram of the components of a typical secure smart card microcontroller.

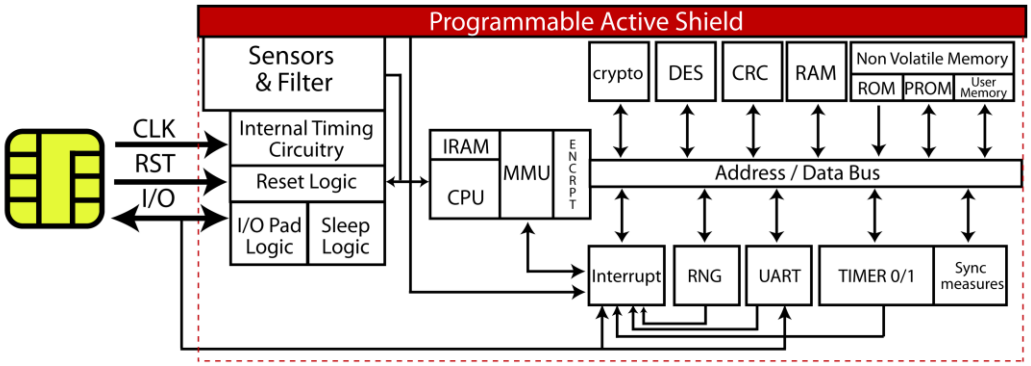


Figure 2. Components of a Typical Secure Smart Card Microcontroller

All IC components provide some aspect of protection against attacks. The following describes how the different components contribute to the security of the IC.

- A **programmable active shield** covers the entire IC and is equipped with signal layers that detect attempts to probe or force internal modules or signal lines.
- A number of **sensors** are built into secure microcontrollers to thwart fault or invasive attacks, including:
  - Low and high frequency sensors for the internal clock
  - Sensors and filters for the external clock
  - External high and low voltage sensors
  - Internal voltage sensors
  - Temperature sensors
  - Peak voltage sensors
  - Glitch sensors on internal voltage
  - Light sensors on the IC surface
- Inaccessible **internal timing circuitry** is used for cryptographic and security operations.
- The **central processing unit** (**CPU**) should have proprietary timing to make it difficult for an attacker to determine the operations that the IC is performing.
- The **memory management unit (MMU)** is an optional module that creates a true hardware firewall within the IC, enhancing the security of multi-application smart card operating systems. It does this by preventing applets from accessing important chip resources that should only be controlled by the card operating system. While this feature adds security for multi-application smart card platforms, it may not necessarily provide value for single application cards running either a single fixed application, nor fixed/configurable file system-oriented card operating systems.
- The **memory and processor bus encryption module (ENCRPT)** encrypts and decrypts stored data using specific keys stored in ROM, RAM, and NVM and a proprietary symmetric algorithm. In addition, the RAM bus (connecting the RAM to the processor) can also be encrypted after each chip reset. These measures prevent an attacker from seeing any IC calculations in the clear if the internal operations of the IC are exposed. Critical registers, the crypto module, and other peripherals are also encrypted.

- **The crypto coprocessors (crypto)** are additional processors that execute either symmetric or asymmetric algorithms such as 3DES, AES, RSA and Elliptic Curve Cryptography (ECC). These engines offload the more intensive cryptographic processing from the CPU, and increase security by implementing hardware security countermeasures. Thus, these countermeasures allow the chip to operate more efficiently and more securely.
- The **Data Encryption Standard (DES)** module performs the calculation of DES and triple DES algorithms.
- The **cyclical redundancy check (CRC)** module verifies data integrity by checking the data to see whether an error has occurred during transmission, reading, or writing. CRC calculations are standardized in the protocol layer; ISO/IEC 7816 for contact smart cards, and ISO/IEC 14443 for contactless smart cards (with coding examples showing how host systems should implement them provided in their appendices).
- The **non-volatile memory (ROM, PROM and user memory)** data are encrypted to prevent an attacker from seeing data as clear text if the data is extracted from the IC.
- **Data bus encryption**. The data that is transmitted along the bus is encrypted, making it difficult for an attacker to determine what is being transported on the bus. All data transmitted to and from security-relevant, special-function registers should be encrypted across the bus. The bus can also scramble addresses being carried and transmitted, making the address scheme more obscure to an attacker.
- A high quality, true **random number generator (RNG)** is the basis of many cryptographic protocols and is also used in conjunction with software to harden cryptography against Differential Power Analysis (DPA) and Simple Power Analysis (SPA). The RNG can be used to create randomly different and false wait states that confuse the attacker when they are attempting to analyze the power consumption of the chip.

  Most importantly, high quality random numbers protect keys when appropriately used in mutual authentication and encryption. In these applications, random numbers are encrypted, exchanged and then eventually used as the basis of session keys guarding transactions. True random numbers are not feasibly guessed by attackers and therefore maximize the strength of the cryptography used.
- A **current masking device** unit scrambles current consumption by performing dummy access operations in memory (ROM, XRAM, and NVM). As a result of scrambling, the current consumption of the actual program flow is hidden. When used in conjunction with the RNG and random wait states, this feature is a powerful countermeasure against power analysis.

3.2.2 Secure Microcontroller Operating System

The secure microcontroller needs an OS to allow it to drive resident applications. The OS is embedded in the IC's ROM during the manufacturing process. The OS not only defines program operations for IC applications, it also includes software security features to counter software attacks and enhance hardware security features. As much as 50 percent of the OS code in a smart card product may be used to support security features. The software developer must be knowledgeable about the IC's architecture so that the OS can be designed to optimize the IC's security module operation.

It is important to note that the speed and performance of a given processor versus another in a specific application should always be judged with each running a secured OS and secured applet/application (independently verified by a trusted third party lab) to ensure one does not have the advantage of running without security.

4 Card Edge Interface Security

As a secure component, a smart card does not expose its internal behavior. Its behavior is only seen at the interface and its available functions are described in what is called as a "card edge," or the set of command-response pairs exchanged at the interface.

ISO/IEC 7816 defines the set of interoperable commands smart cards can use. It also provides for non-interoperable commands which are required for some application functions. The APDU command set defined by ISO/IEC 7816 is richer than one card really requires, by allowing very different data structures to be used by smart card applications, not all of which are required in the same card. For example, native cards with file systems can use binary files or record-oriented files or data objects contained in files. The files can be identified by two methods: one in which they have a two-byte identifier; another in which they have a five-bit-only identifier. (Files may also have both.)

The result is that the APDU commands for interoperability that are defined by ISO/IEC 7816 allow client software to know how each command should behave (and be used). Unfortunately, ISO/IEC 7816 does not define coherent subsets of such command sets, and each operating system, as well as each applet, is able to pick whatever command it likes and not implement the others. This creates all types of card edges, which are different among manufacturers and do not provide interoperability among cards.

Interoperable card edges are defined by application or industry specifications. This has been done for all major applications, such as GSM (ETSI EN 300 812), EMV (EMV v4.1 Book 1 ICC to Terminal Interface), e-Passport (ICAO 9303), and many others.

Some standards attempt to limit the number of APDU commands that a card edge should use, but without defining the data structure and security architecture of the card using these commands, such efforts are futile.

Commands exchanged at the interface may be protected by a process called "secure messaging," described in ISO/IEC 7816 Part 4. By using a session key to cipher or to sign the information exchanged, secure messaging provides the application with integrity, confidentiality, or both security services for commands and data presented at the interface.

ISO/IEC 7816 Part 3 defines the format for all commands. The APDU consists of the following elements:

- **Class byte**: Used to indicate if the command is ISO-compliant or proprietary, if the command is in clear text or uses some cryptographic protection and which logical channel the command is for. (Up to four different processes can be active at a given point in time into a card.)
- **INS byte:** Used to indicate which command is presented to the card at the interface. When the class byte is indicated as proprietary, the command is defined by the application specification itself and not ISO.
- **P1-P2:** Two bytes used as parameters for the command. Their meaning varies depending on the command code.
- **Lc:** Length of the command data field (if present) that is provided as part of the command information to the card. It may be absent.
- **Command data field:** Value of the data (if any) provided to the card as part of the command.
- **Le**: Length expected by the client for the response data field (data sent back in response to the command by the card). The value could be 00 (card to provide the length in the response) or absent (no response expected).
- **Response data field:** Value of the data (if any) sent back by the card in its response to the command. Some commands do not have any data in the response.
- **Status bytes (SW1-SW2):** Two bytes provided by the card when the command has finished its processing. They indicate if the command execution was successful (e.g., SW1-SW2 = '90-00') or if an error happened (e.g., SW1-SW2 = '6D-00' instruction code [INS] not supported).

Table 2 shows examples of card edge interfaces for several major applications. All of these examples have commands which are ISO/IEC 7816-compliant, as well as some commands which are specific to the application requirements and are not described in the common ISO command set.

Application	Commands	
EMV Cards	APPLICATION BLOCK	GET DATA
	APPLICATION UNBLOCK	GET PROCESSING OPTIONS
	CARD BLOCK	INTERNAL AUTHENTICATE
	EXTERNAL AUTHENTICATE	PIN CHANGE/UNBLOCK
	GENERATE APPLICATION CRYPTOGRAM	READ RECORD
	GET CHALLENGE	SELECT
	APPLICATION BLOCK	VERIFY
PIV Cards	CHANGE REFERENCE DATA	PUT DATA
	GENERAL AUTHENTICATE	RESET RETRY COUNTER
	GENERATE ASYMETRIC KEY PAIR	SELECT
	GET DATA	VERIFY
ePassports (ICAO 9303)	SELECT FILE	GET _CHALLENGE
	READ BINARY	EXTERNAL_AUTHENTICATE
		PSO_MSE
		PSO_CDS
		VERIFY_CERTIFICATE
GSM SIM Cards	CHANGE CHV	TA11/12 ALGORITHM
	DISABLE CHV	TA21/22 ALGORITHM
	ENABLE CHV	TA32 ALGORITHM
	GET RANDOM	TA41/52 ALGORITHM
	GET RESPONSE	TA41/TA82 ALGORITHM
	INVALIDATE	TA71 ALGORITHM
	READ BINARY	TB4/TE ALGORITHM
	READ KEY	UNBLOCK CHV
	READ RECORD	UPDATE BINARY
	REHABILITATE	UPDATE RECORD
	SEEK	VERIFY CHV
	SELECT	
	STATUS	

Table 2. Example Card Edge Interfaces

5 Security Implications of Contact and Contactless Interfaces

For some time, the perception was that contactless cards were less secure than contact cards. However, this is not necessarily the case. A contactless card is inherently as secure as a contact card—the same security features are designed into a secure contactless microcontroller as a contact device. One comprehensive study[17] concludes that "contactless technology is not fundamentally more vulnerable than contact technology but specific constraints and threats have to be taken into account and should be solved at the application level." A position paper[18] published by Eurosmart[19] in 2007 also concluded that: "Secure contactless smart card technology provides the same level of security as secure contact smart cards. They use smart card secure microcontrollers with physical security features to protect from tampering and cloning."

Dual-interface secure ICs must be designed so that one mode of operation cannot pose a security risk by leaking information to the other. For this reason, some applications are designed so that both modes cannot be operational at the same time. However, application requirements are emerging that require simultaneous operation so that, for example, a mobile phone can continue with voice communication while also being used to pass through a transit gate. Dual operation does not necessarily pose a security threat. However, the requirements for dual operation must be specified at the design stage.

There has been much media coverage about eavesdropping on RFID devices and contactless payment cards. Most of this coverage confuses RFID technology and secure contactless smart card technology. While both use RF, the former is designed with only limited security and can be read from a long distance, whereas the latter is designed to be secure and read only over a very short distance (a maximum of 10 cm).

Some attacks are now aimed specifically at the contactless interface:

- Eavesdropping, in which an attacker attempts to listen to a valid contactless card using an alternative reader
- Unwanted activation, which is similar to eavesdropping, in which the attacker attempts to activate a genuine contactless card without the card owner's knowledge
- Denial of service, in which the attacker tries to interfere with RF transmissions so that the system does not work and transactions cannot be completed correctly
- Man in the middle, in which a fake reader captures data by intercepting transmissions and relays the information to a fake contactless card by an alternative communication channel, such as an ultra high frequency (UHF) link, which then communicates with an alternative genuine reader

Such attacks can be thwarted with good system design that uses strong authentication and dynamic cryptography. As concluded by Eurosmart, "The use of smart card contactless technology allows secure management of stored and transmitted data using strong encryption, random challenge, access control through authentication and therefore provides countermeasures to defend against the attacks described."[20]

Good security design takes into account the security requirements and limitations of an application at the outset and identifies what risks are acceptable. As is true for all attacks and threats, countermeasures can be implemented, some of which may incur additional costs or be less convenient for users. For

[17] Helena Handschuh, "Contactless Technology Security Issues, Smart Card Security," Information Security Bulletin, Volume 9, April 2004.
[18] "Understanding Secure Contactless Device versus RFID Tag," Eurosmart, http://www.eurosmart.com/images/doc/technical-documents/eurosmart-white-paper-rfid_oct07.pdf
[19] Eurosmart is a non-profit organization located in Brussels that is committed to expanding the world's smart card market, developing smart card standards and continuously improving quality and security applications. Additional information can be found at http://www.eurosmart.com.
[20] Eurosmart, op.cit.

example, a contactless card can be protected by enclosing it in a protective metal sleeve, but then the card must be removed from the sleeve for use. Many countermeasures ultimately involve a tradeoff.

In both contact and contactless environments, it is important to remember that the smart card is only one part of the system. Just as the software within the card can compensate for limitations in hardware (and vice versa), system security measures external to the card can strengthen the security of the overall application.

6 Cryptography and Public Key Infrastructure

This module presents an overview of the world of cryptography, cryptographic systems, public key infrastructure (PKI) and smart cards leveraging these technologies.

The reader will acquire a basic understanding of:

- The reasons for using cryptography
- The data/information that is protected and the mechanisms used to protect
- Cryptographic methods and protocols
- Cryptographic algorithms
- Cryptographic strength and forward security issues
- Protection of keying material by smart cards
- PKI and establishing trusted infrastructures
- Definitions of certificate policy, certificate practices, key recovery
- Deployment strategies for PKI
- Application use cases for cryptography

6.1 Cryptography

> **Note:** The outline of this material on cryptography follows some sections of material from Peter Gutmann's tutorial on cryptography. In this section, many of the figures are extracted from his content. If broader and more extensive information on cryptography is needed, his tutorial published on the web[21] is an extremely useful reference. In addition, a good reference on cryptography is Bruce Schneier's book "Applied Cryptography."[22]

A secret is hard to communicate and have it remain a secret. A secret cannot be made out of a non-secret. So systems use cryptographic methods to provide confidentiality, integrity, authentication and non-repudiation. Specifically, the following are definitions that support cryptographic methods:

- Confidentiality – provides protection from disclosure to unauthorized persons
- Integrity – maintains data consistence (from sender to recipient)
- Authentication – provides assurance of identity of a person or originator of data
- Non-repudiation – ensures the originator of a message cannot deny it later

Various cryptographic techniques and protocols are used to ensure these attributes of secure messaging work effectively. If cryptographic methods are misused, the system may look secure, but it truly is not. The following sections will discuss the foundation of cryptographic tools and how they are used in combination to ensure these attributes are achieved.

In addition to these security requirements, there are operational and usability requirements, including availability, access control and appropriate combinations of the two. Specifically, these requirements are defined as follows:

- Availability – defines the need for legitimate users to have access when they need it
- Access control – ensures that unauthorized users are kept out and are not able to break in
- Combinations – uses user authentication for access control purposes and combines non-repudiation with authentication

Security threats drive the use of cryptography. Threats can be:

[21] Peter Gutmann's tutorial materials are available at http://www.cs.auckland.ac.nz/~pgut001/tutorial/index.html
[22] Bruce Schneier, Applied Cryptography: Protocols, Algorithms and Source Code in C (Wiley 1996)

- Information disclosure or leakage – a data breach (sometimes dramatically published in the press)
- Integrity violations – alteration of information in transit to the benefit of the attacker
- Masquerading – enabling an attacker to look legitimate
- Denial of service – disruption of daily operations
- Illegitimate use – gaining access and turning an individual's personal computer (PC) into a zombie server
- Generic threats – backdoors, trojans and insider attacks

Security threats define the business risk. Figure 3 shows a simplified view of attacks that are used. Passive attacks can only observe communications or data. Active attacks can actively modify communications or data.

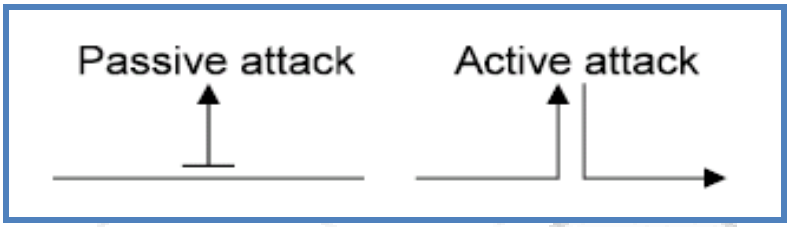


Figure 3. Types of Security Attacks

To mitigate threats and attacks, cryptography uses three basic building blocks.

- Symmetric encryption algorithms are used to provide confidentiality, as well as authentication and integrity protection.
- Asymmetric algorithms provide capabilities for digital signatures and encryption and are used to provide authentication, integrity protection and non-repudiation.
- Checksums and hash algorithms are used to provide integrity protection (and a lightweight mechanism for authentication).

One or more security mechanisms are combined to provide a security service.

Figure 4 shows a constructive example of how the security mechanisms fit together. SSL, for example, relies on cryptographic methods for signatures, encryption and hashing. These mechanisms use specific cryptographic algorithms, such as DSA and RSA (two of three available asymmetric cryptography algorithms, the third being elliptic curve cryptography or ECC), DES for symmetric encryption, and secure hashes provided by the cryptographic algorithms SHA1 or MD5.

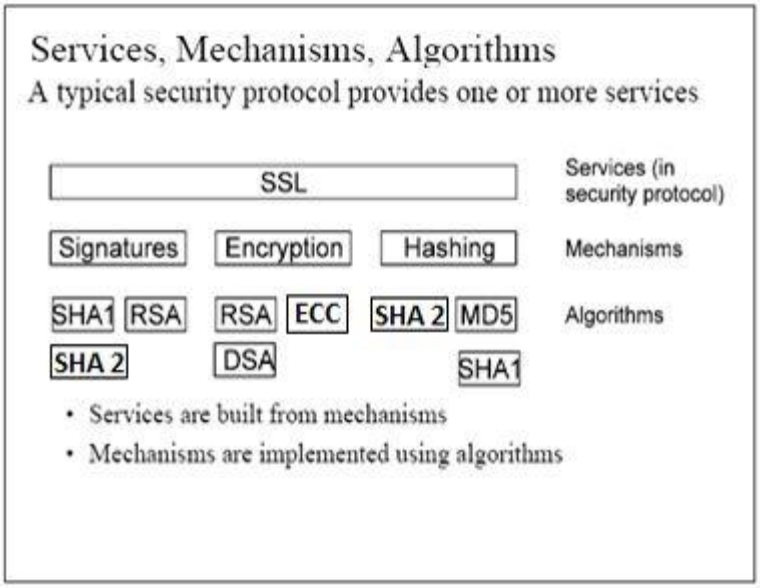


Figure 4. Example of Services, Mechanisms and Algorithms

The next sections define the tools of cryptography that are used in Figure 4.

6.1.1 Cryptographic Methods

Conventional encryption is a shared secret model that is symmetric (i.e., symmetric encryption). With symmetric encryption, the key used to encrypt the information by the sender is the same key used to decrypt the information by the recipient. Figure 5 shows the basic flow. The challenge in this model is the number of people that may need to have a copy of the key to make the data useful in a real working system. The greater the number of people having the key, the less likely it will stay secret. If too many people have the key, it becomes a non-secret and the data protected by that particular key is easily compromised.

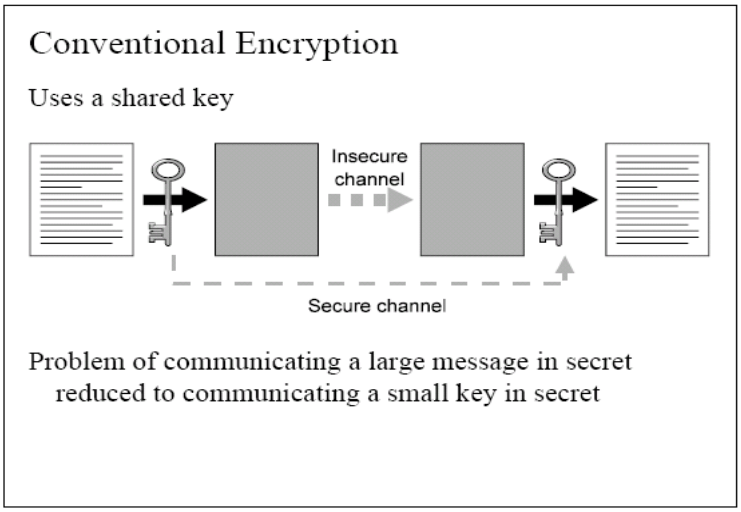


Figure 5. Conventional Encryption

This model also has a tremendous problem with key distribution. Figure 5 shows a secure channel used to distribute the keys, yet an insecure channel for the encrypted message. In many symmetric schemes, this secure channel is “out of band.” The encrypted information is sent electronically, but the encryption

key is sent via secured postal mail. Establishing the secure channel for key distribution is considered to be a serious problem.

Public key encryption (also called asymmetric encryption) was invented in 1975 and was a breakthrough that addressed the key distribution challenges. As shown in Figure 6, a key pair, one public and one private, is established. Only one individual should ever possess the private key and it must be kept secret. The public key is shared with anyone. Anyone can encrypt with the public key, but only the one person with the private key can decrypt it. This process can also be reversed where anyone can encrypt a message using someone's public key and only the owner of the private key can decrypt it. This dramatic discovery initiated the world of public key infrastructure (PKI).

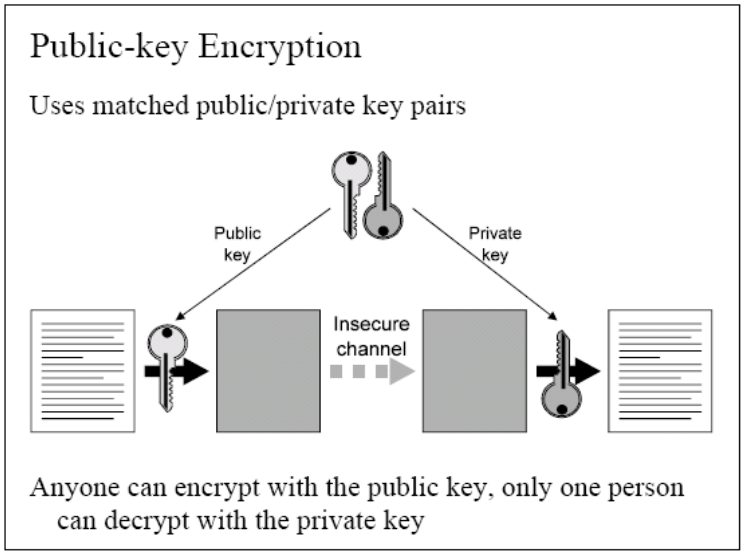


Figure 6. Public Key Encryption

Figure 6 shows the very basic model of asymmetric (public key) encryption. In this figure, a message is being sent to a specific individual. Using that individual's public key, the message is encrypted. (This message is very often a symmetric key that will be used for other cryptographic functions.) The encrypted message is then sent to the owner of the private key. The owner of that key can then decrypt the message. This process provides a means to ensure that *only* the private key owner will see the message.

Hash functions, illustrated in Figure 7, are useful tools to reduce the size of messages to a manageable level. They provide a means to check the integrity of the data. When the data is sent with its appropriate hash value, the receiver can check that the data is unaltered by regenerating the hash value upon receipt. If the resulting hash values are not identical, then something is suspect in the data (or the hash value) as received.

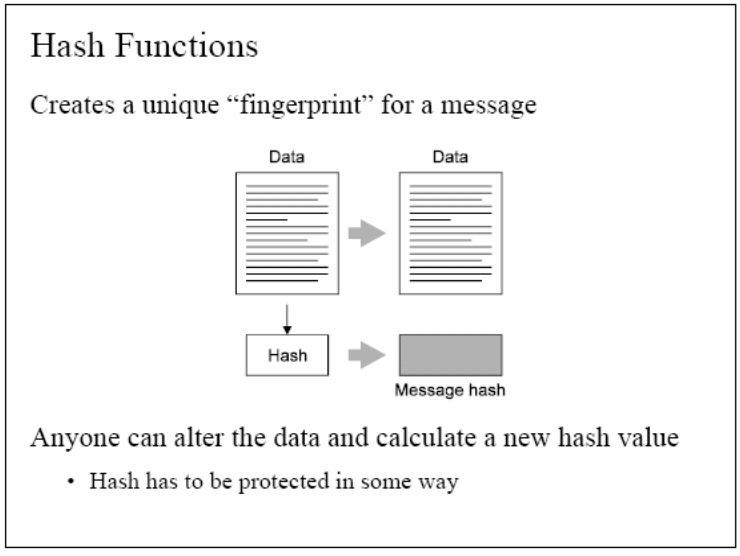


Figure 7. Hash Functions

Message authentication codes

(MACs) build upon the concept of a hash, but add a security element of an encryption key. Figure 8 shows the basic process. The hash algorithm is used to reduce the size of the data to a reasonable size. This algorithm is then continued with an additional input (the MAC key). Now the MAC is specific to that particular key. Only holders of the MAC key are able to recreate the MAC for verification purposes. In this model, the MAC key is a shared secret that must be available to the relying party so that they can verify the integrity of the MAC. The primary goal of a MAC is a lightweight, high performance way to communicate who generated the MAC. As with any symmetric method, keeping the key secret is required to ensure that the code is useful.

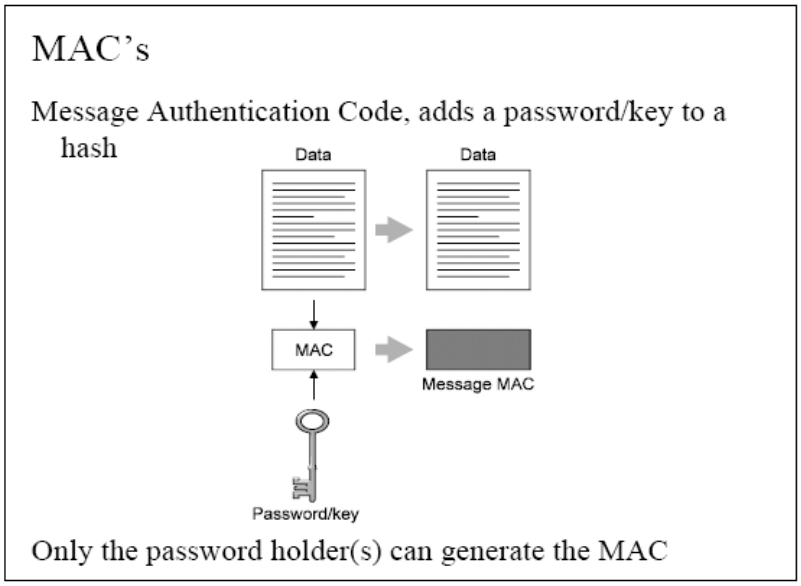


Figure 8. Message Authentication Codes

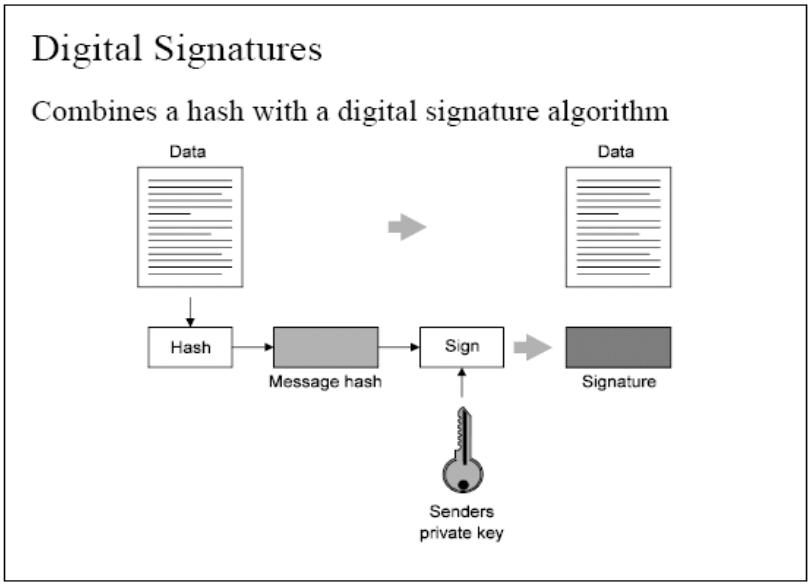


Figure 9. Digital Signatures

Digital signatures start to layer together these basic concepts for a more powerful and useful result. A digital signature (as shown in Figure 9) leverages a hash function to reduce the size of the message to be

encrypted to a manageable size. The sender's private key is used to encrypt the hash, creating a signature. The data along with the signature are then sent to any relying party (i.e., the recipient). The relying party can then use the public key of the sender to decrypt the hash value, re-hash the original document, and compare the two hash values. If they match, the signature is verified. As such, any verifier will know this message was sent by the individual possessing the private key and that the data is unaltered. This process is shown in Figure 10.

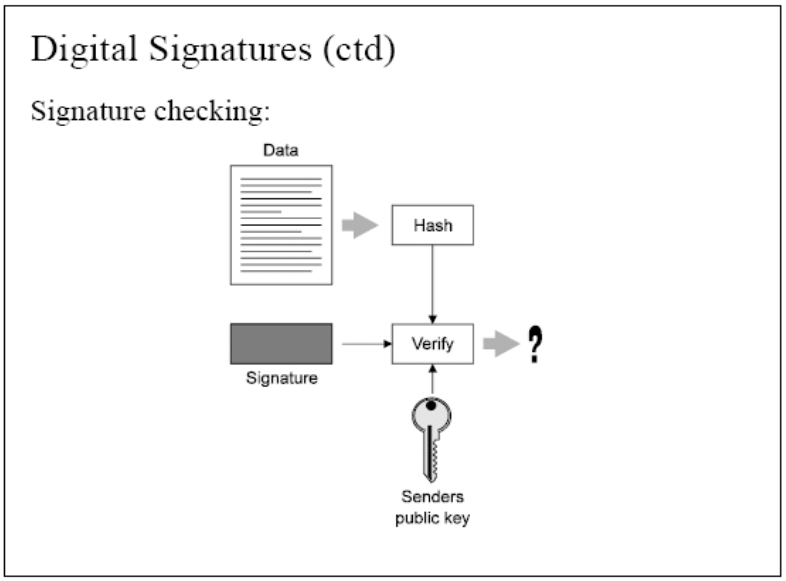


Figure 10. Signature Checking

Figure 11 illustrates combining the use of a public key with the symmetric cipher to send an encrypted message to an individual. S/MIME email uses these protocols. A symmetric key is used to do the bulk encryption of the data. (Symmetric encryption is about 1000 times faster than public key encryption.) To send this data securely to the recipient, the symmetric key (a very small object) is encrypted using the public key of the recipient. The encrypted key and the encrypted message are then sent to that recipient.

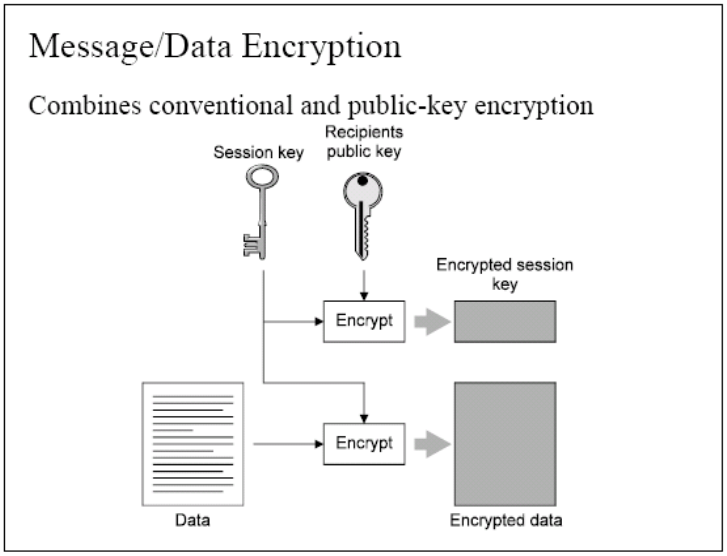


Figure 11. Secure Message Encryption

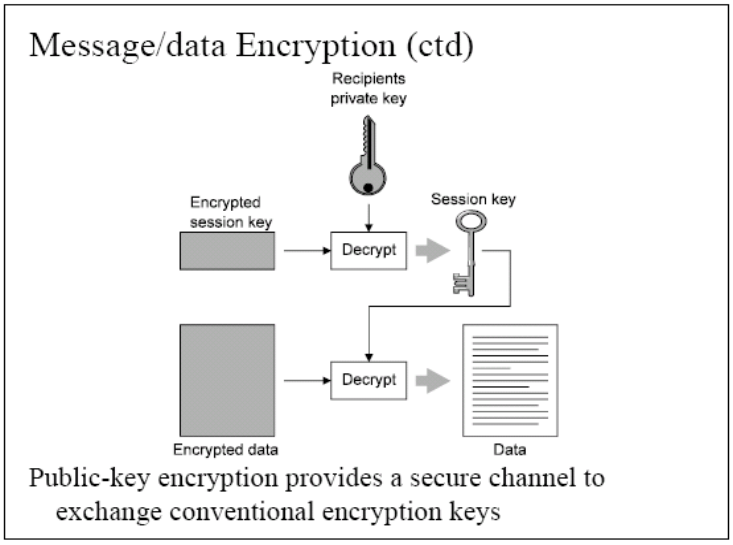


Figure 12. Secure Message Decryption

The corresponding recipient actions to decrypt the message from the sender are shown in Figure 12. In this model, the recipient's private key is used to decrypt the symmetric encryption key (i.e., a session key or ephemeral key). This symmetric encryption key is then used to decrypt the message so that the recipient can now see the original message. This process provides confidentiality. Because the message was encrypted, it is computationally infeasible to decrypt (to get back to the cleartext message) if it is intercepted in transmission.

Additional protocol work is required to ensure integrity, non-repudiation and authentication of the sender. This would involve, for example, the sender adding unique information to the message (e.g., date and time, or sender's transaction counter), then signing a hash of the entire message and providing a public key certified by a trusted verifiable path.

Using these basic building blocks, industry has developed a wide range of protocols and services to support reliable messaging and commerce over the Internet. Figure 13 shows some of the tools that have been created for these purposes.

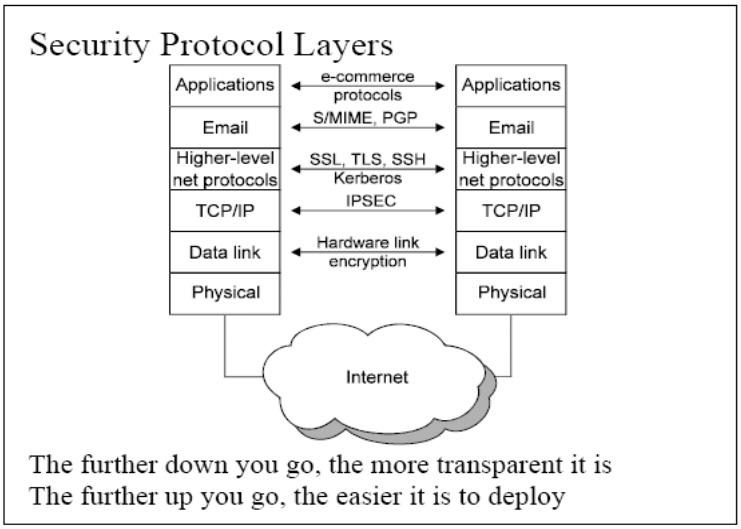


Figure 13. Security Protocol Layers

6.1.2 Cryptographic Algorithms[23]

Many cryptographic algorithms and implementations are available to meet the needs described in Section 6.1. In general, many algorithms can manage hash functions and symmetric ciphers, there are only three successful asymmetric algorithms: RSA, DSA and ECC. Meeting the mathematical requirements and principles for asymmetric, public/private key cryptography is a challenging problem.

This section will discuss briefly the algorithms that are used for government and many financial and mobile telecommunications applications.

This section includes content from a number of U.S. government standards that have defined how cryptography is implemented within government applications. While defined for the U.S. government, the descriptions below are valid for the use of cryptography in general.

Two critical documents defining cryptography are NIST Special Publication (SP) 800-21[24] and NIST Special Publication 800-57[25]. SP800-21 Section 3.1 provides an excellent definition of cryptography in a general sense, and the rationale for cryptographic applications to protect data. It then establishes the categories of cryptographic algorithms that will be further discussed in this module.

> Cryptography is a branch of mathematics that is based on the transformation of data and can be used to provide several security services: confidentiality, data integrity, authentication, authorization and non-repudiation. Cryptography relies upon two basic components: an *algorithm* (or cryptographic methodology) and a *key*. The algorithm is a mathematical function, and the key is a parameter used in the transformation.
>
> A cryptographic algorithm and key are used to apply cryptographic protection to data (e.g., encrypt the data or generate a digital signature) and to remove or check the protection (e.g., decrypt the encrypted data or verify the digital signature). There are three basic types of Approved cryptographic algorithms: cryptographic hash functions, symmetric key algorithms and asymmetric key algorithms:
>
> - Cryptographic hash functions do not require keys (although they can be used in a mode in which keys are used). A hash function is often used as a component of an algorithm to provide a security service.
> - Symmetric algorithms (often called secret key algorithms) use a single key to both apply the protection and to remove or check the protection.
> - Asymmetric algorithms (often called public key algorithms) use two keys (i.e., a key pair): a public key and a private key that are mathematically related to each other.
>
> Random number generators (RNGs) are required for the generation of cryptographic values (e.g., keys).

6.1.2.1 Hashes

Hashes are defined as follows:[26]

[23] Sources: This section extracts content from NIST Special Publications to define cryptographic algorithms.
[24] NIST SP800-21-1, *Guideline for Implementing Cryptography In the Federal Government, Second Edition*, December 2005
[25] NIST SP800-57, *Recommendation for Key Management, Parts 1 and 2*, August, 2005
[26] NIST SP800-21-1, *Guideline for Implementing Cryptography In the Federal Government, Second Edition*, December 2005

> A hash function produces a short representation of a longer message. A good hash function is a one-way function: it is easy to compute the hash value from a particular input; however, backing up the process from the hash value back to the input is extremely difficult. With a good hash function, it is also extremely difficult to find two specific inputs that produce the same hash value. Because of these characteristics, hash functions are often used to determine whether or not data has changed.

FIPS 180-4, *Secure Hash Standard*[27] defines five hashing algorithms : SHA-1, SHA-224, SHA-256, SHA-384 and SHA-512 (SHA: Secure Hash Algorithm). New attacks have also been identified for SHA-1 and SHA-1 is no longer recommended to be used for digital signatures in new systems for Federal applications. SHA-1 will be replaced in the U.S. Federal government by SHA-256 in all digital signature applications at the end of 2010.

6.1.2.2 Symmetric Algorithms

Symmetric algorithms are defined as follows:[28]

> Symmetric key algorithms (often call secret key algorithms) use a single key to both apply the protection and to remove or check the protection. For example, the key used to encrypt data is also used to decrypt the encrypted data. This key must be kept secret if the data is to retain its cryptographic protection. Symmetric algorithms are used to provide confidentiality via encryption, or an assurance of authenticity or integrity via authentication, or are used during key establishment.

Symmetric algorithms commonly used in applications include:

- **Data Encryption Standard (DES)**. The Data Encryption Standard (DES) is a block cipher (a form of shared secret encryption) that was selected by the National Bureau of Standards as an official Federal Information Processing Standard (FIPS) for the United States in 1976 and which has subsequently enjoyed widespread use internationally. It is based on a symmetric-key algorithm that uses a 56-bit key. DES is now considered to be insecure for many applications.[29]

  DES was withdrawn as an approved algorithm for the U.S. Federal government in 2005. The strength of this algorithm is no longer sufficient to protect Federal information since it uses 56-bit keys. Although the algorithm itself is secure, the key length is too short and exhaustive key attacks are feasible with today's technology.

- **Triple Data Encryption Algorithm (TDEA)** (also referred to as Triple DES (or 3DES)). Triple DES (3DES) is the common name for the Triple Data Encryption Algorithm (TDEA) block cipher, which applies the Data Encryption Standard (DES) cipher algorithm three times to each data block. Because of the availability of increasing computational power, the key size of the original DES cipher was becoming subject to brute force attacks; Triple DES was designed to provide a relatively simple method of increasing the key size of DES to protect against such attacks, without designing a completely new block cipher algorithm.[30,31]

  NIST SP 800-67 Rev 1[32] provides recommendations for use of the TDEA algorithm. Two keying options are approved: 2TDEA (using two 56 bit keys) and 3TDEA (using three 56 bit keys). 2TDEA

[27] FIPS 180-4, *Secure Hash Standards (SHS),* May 2012
[28] NIST SP800-21-1, *op. cit.*
[29] Source: http://en.wikipedia.org/wiki/Data_Encryption_Standard
[30] Source: http://en.wikipedia.org/wiki/Triple_DES
[31] Additional information on TDEA can be found in NIST Special Publication 800-67 Rev 1 Recommendations for the Triple Data Encryption Algorithm (TDEA) Block Cipher (http://csrc.nist.gov/publications/nistpubs/800-67-Rev1/SP-800-67-Rev1.pdf) and ISO/IEC 18033-3:2005 Information technology — Security techniques — Encryption algorithms — Part 3: Block ciphers (http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=37972).
[32] NIST SP 800-67 Rev. 1, *Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher*, January 2012

provides 80 bits of key strength. Only 3TDEA will be supported for U.S. Federal government use until 2030, providing 112-bit key strength.

- **Advanced Encryption Standard (AES)**. AES was developed to replace DES and is the preferred algorithm for all new products. It offers key strengths of 128, 192 and 256 bits. FIPS 197[33] provides the full specification for AES. As with DES, AES is in use worldwide. In developing AES, NIST used a standardization process to evaluate and select the final design; 15 designs were presented during the process. The selected algorithm, the Rijndael algorithm, was developed by two Belgian cryptographers, Joan Daemen and Vincent Rijmen.[34]

6.1.2.3 Asymmetric Algorithms

As described in Section 6.1.1, the asymmetric encryption algorithm was described in 1976 by Whitfield Diffie and Martin Hellman and addresses the key distribution challenges associated with symmetric encryption. With an asymmetric algorithm, a key pair, one public and one private, is established. Only one individual should ever possess the private key and it must be kept secret. The public key is shared with anyone. Anyone can encrypt with the public key, but only the one person with the private key can decrypt encrypted data. This process can also be reversed where anyone can encrypt a message using someone's public key and only the owner of the private key can decrypt it.

Asymmetric key algorithms solve the difficult problem of protecting the secret (shared) key of symmetric algorithms, but they created another issue of their own – they require the public key that is provided in a public/private key pair to be certified by a trusted authority. As such, asymmetric algorithms must be used in a public key infrastructure (PKI) to provide trust in the protection mechanisms. (See section 6.3.)

Asymmetric algorithms used in applications include:

- **RSA algorithm**. The RSA algorithm was described by Ronald L. Rivest, Adi Shamir and Leonard Adelman in 1978 and is widely used worldwide. RSA includes both public and private keys that are generated from two large prime numbers. RSA keys must be sufficiently long to ensure security. While 1024 bit keys are currently in use, key sizes of 2048 bits and higher are now also used and being mandated for some applications.[35]
- **Digital Signature Algorithm**. The Digital Signature Algorithm (DSA) is a United States Federal Government standard for digital signatures. It was proposed by the NIST in August 1991 for use in their Digital Signature Standard (DSS), specified in FIPS 186, adopted in 1993.[36]
- **Elliptic curves**. Elliptic curve cryptography (ECC) is an approach to public key cryptography based on the algebraic structure of elliptic curves over finite fields. The use of elliptic curves in cryptography was suggested independently by Neal Koblitz and Victor S. Miller in 1985.[37] Using ECC requires less computational power than RSA and equivalent cryptographic strength can be achieved with shorter keys.[38] ECC is still only used to a limited extent.

6.1.2.4 Asymmetric Algorithms and Digital Signatures

Digital signatures are described as follows:[39]

---

[33] FIPS 197, *Advanced Encryption Standard*, November 2001
[34] Source: http://en.wikipedia.org/wiki/Advanced_Encryption_Standard
[35] For example, NIST SP 800-78-4 specifies that the U.S. FIPS 201 PIV Card use 2048 bit RSA keys for the digital signature key and key management key as of 12/31/2008 and for the PIV authentication key and card authentication key after 12/31/2013.
[36] Source: http://en.wikipedia.org/wiki/Digital_Signature_Standard
[37] Source: http://en.wikipedia.org/wiki/Elliptic_curve_cryptography
[38] Rankl, Effing, Smart Card Handbook, Fourth Edition, John Wiley & Sons, 2010
[39] NIST SP800-21-1, *Guideline for Implementing Cryptography In the Federal Government, Second Edition*, December 2005

Digital signatures authenticate the integrity of the signed data and the identity of the signatory. A digital signature is represented in a computer as a string of bits and is computed using a digital signature algorithm that provides the capability to generate and verify signatures. Signature generation uses a private key to generate a digital signature. Signature verification uses the public key that corresponds to, but is not the same as, the private key to verify the signature. Each signatory possesses a private and public key pair. Signature generation can be performed only by the possessor of the signatory's private key. However, anyone can verify the signature by employing the signatory's public key. The security of a digital signature system is dependent on maintaining the secrecy of a signatory's private key. Therefore, users must guard against the unauthorized acquisition of their private keys.

SP800-57 Part 1 Rev 3[40] provides guidance that defines the algorithms and how they can be used.

SP 800-57, Section 4.2.4 Digital Signature Algorithms

Digital signatures are used to provide authentication, integrity and non-repudiation. Digital signatures are used in conjunction with hash algorithms and are computed on data of any length (up to a limit that is determined by the hash algorithm). [FIPS186-3] specifies algorithms that are approved for the computation of digital signatures. It defines the Digital Signature Algorithm (DSA) and adopts the RSA algorithm as specified in [ANSX9.31] and [PKCS#1] (version 1.5 and higher), and the ECDSA algorithm as specified in [ANSX9.62].

SP 800-57-2, Section 4.2.4.1 DSA

The Digital Signature Algorithm (DSA) is specified in [FIPS186-3] for specific key sizes: 1024, 2048, and 3072 bits. The DSA will produce digital signatures of 320, 448, or 512 bits. Older systems (legacy systems) used smaller key sizes. While it may be appropriate to continue to verify and honor signatures created using these smaller key sizes, new signatures shall not be created using these key sizes.

SP 800-57, Section 4.2.4.2 RSA

The RSA algorithm, as specified in [ANSX9.31] and [PKCS#1] (version 1.5 and higher) is adopted for the computation of digital signatures in [FIPS186-3]. [FIPS186-3] specifies methods for generating RSA key pairs for several key sizes for [ANSX9.31] and [PKCS#1] implementations. Older systems (legacy systems) used smaller key sizes. While it may be appropriate to continue to verify and honor signatures created using these smaller key sizes, new signatures shall not be created using these key sizes.

SP 800-57, Section 4.2.4.3 ECDSA

The Elliptic Curve Digital Signature Algorithm (ECDSA), as specified in [ANSX9.62], is adopted for the computation of digital signatures in [FIPS186-3]. [ANSX9.62] specifies a minimum key size of 160 bits. ECDSA produces digital signatures that are twice the length of the key size. Recommended elliptic curves are provided in [FIPS186-3].

In addition to digital signatures, asymmetric algorithms are critical in solving the key distribution issue of shared secrets as used by symmetric algorithms.

[40] NIST Special Publication 800-57 "Recommendation for Key Management, Part 1 R3 July, 2012

6.1.2.5 Random Number Generators

Random number generators are defined as follows:[41]

> Random number generators (RNGs) are required for the generation of keying material (e.g., keys and IVs). Two classes of RNGs are defined: deterministic and non-deterministic. Deterministic Random Bit Generators (DRBGs), sometimes called deterministic random number generators or pseudorandom number generators, use cryptographic algorithms and the associated keying material to generate random bits; Non-Deterministic Random Bit Generators (NRBGs), sometimes called true RNGs, produce output that is dependent on some unpredictable physical source that is outside human control.
>
> [FIPS186-3] defines a DRBG that may be used to generate random bits for cryptographic applications (e.g., key or IV generation). The DRBG is initialized with a secret starting value, called a RNG seed. An "attacker" with knowledge of the DRBG output should not be able to determine the seed other than by exhaustive guessing.

6.1.3 Cryptographic Strength and Forward Security

"Strength of mechanism" is a cryptographic concept that, in general, is a mathematical analysis that determines the time it would take an attacker to break the cryptosystem. This calculation looks at the computational power required (number of computers of a particular speed, methods, time) for a successful attack. Strength of mechanism is often described in symmetric-equivalent key strength. Table 3 shows this relationship for the algorithms discussed in previous sections.

The development of protocols or cryptographic methods must rely on tables such as this to ensure that each part of a protocol is the appropriate strength. As with any other security mechanism, the strength is only as good as the weakest link. Using an 80-bit symmetric algorithm to encrypt data, then using an ECC 256 bit key to protect it will only yield 80 bits of security for that particular transaction. The equivalent strength required would be AES-128 to ensure that the symmetric cipher would not be the weakest link.

Equivalent Key Strength (bits)	80	112	128	192	256
Symmetric Algorithm	2TDEA	3TDEA	AES-128	AES-192	AES-256
ECC Key Length	161	224	256	384	512
RSA Key Length	1024	2048	3072	7680	15360

Table 3. Equivalent Key Strengths

An operational example of this can be found in NIST SP 800-78.[42] This special publication describes the appropriate key strengths and lengths for all algorithms used for the Federal Personal Identity Verification (PIV) system and PIV-compliant smart cards.

"Forward security is another concept that must be considered. If a document is digitally signed, how long is it expected that no one will be able to break the private key and forge a signature on an altered

[41] NIST Special Publication 800-57, " *Recommendation for Key Management*," Section 4.2.7, March 2007
[42] NIST SP 800-78, *Cryptographic Algorithms and Key Sizes for Personal Identity Verification*, August 2007

document? Consider a presidential signature affirming a law or a signature on a travel expense report. The presidential signature may need to be secure for a minimum of 50 years (when presidential papers are released), whereas the travel expense report may only need a signature to be valid for the next three years (according to financial record-keeping obligations). Forward security must be considered by any application using of cryptography." NIST SP 800-78-3 defines the operational use dates for particular algorithms PIV credential and PIV system application and provides an excellent example of forward security decisions for other applications to consider. (SP 800-78-4 is currently in draft)

6.2 Establishing Trust

Section 6.1 provided the definitions of cryptographic algorithms and started the discussion on their use within protocols and applications to provide assurances around integrity, confidentiality, authentication and non-repudiation. This section discusses the use of cryptography to provide trust models for applications.

6.2.1 Protection of Keying Material

Five models are used to protect keying material. They are:

- **Software**. Software is the most widely used form and provides some amount of trust. Generally speaking, the keying material is encrypted and stored on a PC. A password is required to decrypt the keying material prior to use. The key is subsequently erased from memory (in most applications) when it is no longer required.
- **Hardware security module**. Hardware security modules are *general purpose* cryptographic modules. They provide certified storage of cryptographic keys and commands that support their operational use. The certification programs include FIPS 140 and Common Criteria evaluations.[43]
- **Host security module** or security application module. Host security modules or security application modules are *application-specific* cryptographic modules. They provide certified storage of cryptographic keys and commands that support their operational use. Application-specific behavior is typically fine-tuned for a specific purpose, such as financial transactions by banks. Host security modules are used to support PIN encryption and symmetric processing for verification of financial transactions involving the diversified keys of smart cards. The certification programs include FIPS 140 and Common Criteria evaluations.
- **Protected terminals**. Protected terminals include mobile phone handsets, payment terminals, and set-top boxes. These devices protect transactions and keys in accordance with the issuer's application needs to enable a service.
- **Smart cards**. Smart cards are a mixture of the definition of host and hardware security modules. They provide certified storage of cryptographic keys and commands that support their operational use. Depending on the application, these cards may be general purpose applications or they may be tuned for very specific purposes. PIV credentials tend toward general purpose applications (e.g., PKI support), while financial credentials tend toward application-specific models (e.g., specific diversified keys designed for individual financial transactions). The certification programs include FIPS 140 and Common Criteria evaluations.

Host security modules tend to be highly optimized devices, seeking to support hundreds or thousands of transactions per second in financial environments. Hardware security modules are also highly optimized, but they support general purpose protocols such as SSL, TLS and PKI systems. Smart cards are used in both models to support end user keying material. Smart cards are the best known solution to enable high assurance cryptography and replace software protection of keys for end users.

Both symmetric and asymmetric trust models depend on host or hardware security modules to protect keying material that is critical for a particular trust model. For asymmetric models, private keys must be

[43] Additional information on FIPS 140 and Common Criteria can be found in Section 2.

stored in hardware. For symmetric models, PINs and master keys are stored in hardware. High assurance systems use smart cards for the end user and HSMs for other entities.

6.2.2 Symmetric Mechanisms

Symmetric mechanisms use a wide range of trust models. These include:

- Code books
- One-time passwords
- System wide secrets
- Issuer secrets and diversified keys for end users

Code books and one time passwords (sometimes known as one time pads) are shared ahead of time and define a particular key to use for the symmetric encryption operation. Both parties have access to these books in order to encrypt and decrypt material using the correct keys. (In World War I, for example, this was simply a pad of paper with the "passwords" printed on a single page.) One-time passwords (OTPs) are available electronically using open or proprietary algorithms.[44] Smart cards provide excellent protection and generation methods for OTP models for the end user.

System-wide secret keys are a significant risk to most systems and should be avoided. In these systems, a master secret is retained on servers by the owner of an application. That secret is then "hidden" and "protected" by devices such as smart cards, terminals or software. System-wide secrets should always be avoided.

Financial and mobile phone applications using smart cards and dedicated terminals (handsets and terminals) have developed an excellent solution to avoid putting system-wide secrets in consumers' hands. This model is the use of "diversified keys." For a financial application, the bank holds the master secret key. Using some unique value (such as a smart card chip ID number, bank account number, or terminal ID number), that master secret is hashed with the unique value. The resulting hashed value is then used as the key to calculate a MAC or to encrypt a value using DES or TDEA. When the transaction is received by the issuer, an HSM is used to calculate the value of the key for that particular device and verify the MAC or decrypt the value. Using this technique, the transaction can be authenticated and authorized.

The strength of diversified key schemes is similar to the more general asymmetric models: if one account's card is broken, only that particular account can be attacked. This results in a detectable failure so that the account can be terminated and a new card can be issued to the account holder. Without the diversified scheme, every account would be broken and the attack would require the re-issuance of cards to *every* account holder.

To ensure that other attack models and threat vectors cannot be exploited, financial applications tend to use very controlled specifications to maintain the integrity of secrets and protocols. This safeguard limits the amount of information available to the attacker, making it more difficult. Yet security by obscurity rarely works forever. Eventually, people with enough spare time will announce some attack against a part of the system. It is very important to remember that these attacks rarely break the system using a diversified scheme, rather the attacker gains access to a single account. The issuer can often easily address this attack by simply re-issuing a new card to the account owner.

6.2.3 Asymmetric Mechanisms

The largest body of work on using asymmetric algorithms for trust is the use of PKI, discussed in Section 6.3.

Alternative models using asymmetric mechanisms include:

[44] The Initiative for Open Authentication (OATH) has defined open specifications for one-time password algorithms. Additional information on OATH can be found at http://www.openauthentication.org.

- Account-based models
- Bilateral agreements
- PKI Lite

6.2.3.1 Account-Based Models

Account-based models use smart cards for the protection of a client's private keys. The corresponding public key is registered within the account record (in place of a PIN, password or symmetric diversified key scheme). This reduces the complexity of account management at the account records, as the public keys to not need application-specific HSMs. The account-based model is illustrated in Figure 14.

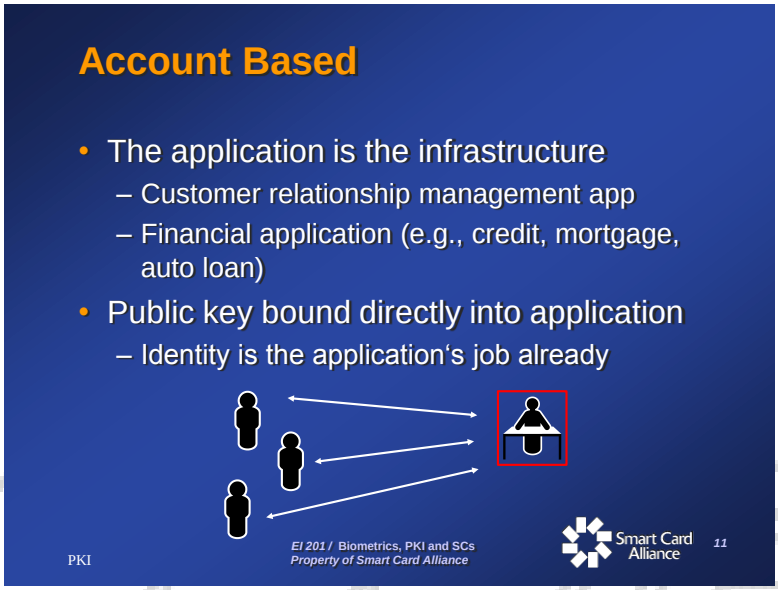


Figure 14. Account-Based Model

This model specifically assumes that the application already knows the customer. As an example, consider a financial application.

- The bank received the customer's application for a credit card account.
- It reviewed and approved the application.
- It created an account record for the customer.
- It bound the customer's public key to that account record (in place of a PIN).
- It issued a smart card to the customer with the corresponding private key for that account.

As the account holder, when the customer uses the private key on the smart card, the customer is the only individual who possesses that card. The bank simply looks up the account record to verify the transaction with the customer's public key. Risk of compromise of a record by insider attack (decrypted PIN or secret key) is minimized. This system is also highly scalable.

6.2.3.2 Bilateral Agreements

Bilateral agreements are a lightweight model for cross certification. Two parties with existing PKI infrastructures cross-certify themselves to establish trust between the parties. This is illustrated in Figure 15.

This model may work well between two organizations; however, it is not scalable and presents specific policy mapping problems if additional organizations seek to join. These models have largely been replaced by PKI trust models using bridge certificate authorities that are discussed in the next section.

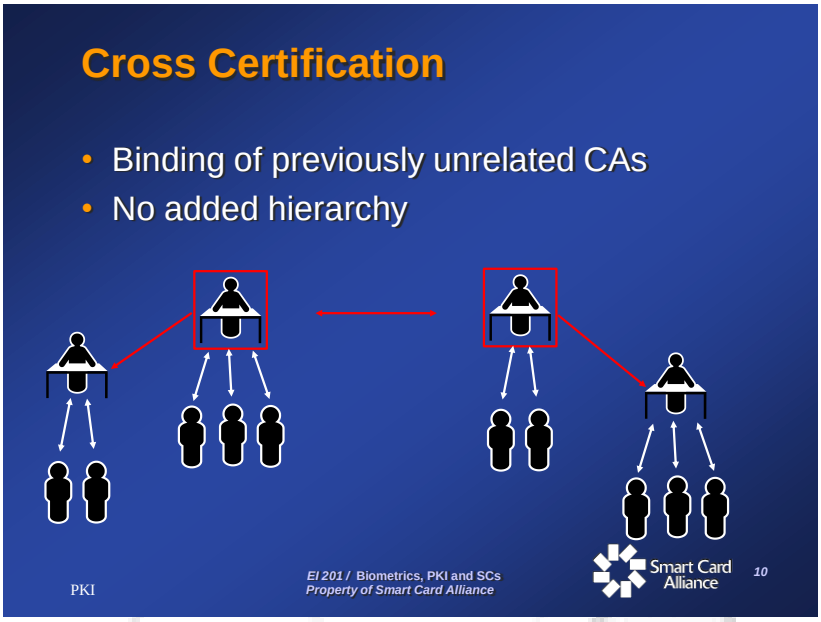


Figure 15. Bilateral agreements

6.2.3.3 PKI Lite

The best known model of a PKI Lite implementation is International Civil Aviation Organization (ICAO) Machine Readable Travel Documents[45] (MRTD) infrastructure. MRTDs are better known as passports and visas.

For ePassports (and visas), integrity is the issue. Issuing countries uses PKI to *sign* the information in the ePassport, certifying that the document came from a particular country of origin. The public keys for all countries are circulated around the world using a directory managed by ICAO. Any country that receives an ePassport can simply verify the integrity of the data on that ePassport by verifying the signature(s) on the ePassport using the issuing country's public key(s). This provides a significant optimization in the use of PKI and is very similar to the account-based model. The key difference is that the verification is performed by the relying party (the country the ePassport holder wishes to enter), not by the issuing country of the ePassport.

6.3 Public Key Infrastructure (PKI)

PKI is an infrastructure that has a principal mission: transitive trust. As an example, PKI is used to ensure that Alice can trust Bob, even if Alice has never met Bob before. This is called transitive trust. Certificate authorities (CAs) within a PKI are trusted third parties. The CA issues digital certificates to Alice and Bob so trust can be established. A digital certificate binds an asymmetric public key to identity information about Alice, under a given PKI policy. That digital certificate, presented by Alice to Bob, along with Alice's signature on a message, enables Bob to trust Alice. Bob can verify Alice's signature and also query the CA to ensure that the digital certificate is still valid. If both check out, Bob knows he was receiving a specific, confirmed message from Alice.

A key element of a PKI is the ability to check that a digital certificate is valid. This can be implemented in a number of ways, including with a certificate revocation list (CRL) or Online Certificate Status Protocol (OCSP) responder.

- A CRL is a list of certificates (or more specifically, a list of serial numbers for certificates) that have been revoked or are no longer valid, and therefore should not be relied upon. A CRL is

[45] MRTDs are better known as ePassports and visas.

generated and published periodically by the certificate authority (CA) that issued the corresponding certificate.[46] When a certificate is used, the relying party would check the CRL to determine the validity of the certificate. Using the example above, Bob would consult the CRL from the CA that issued Alice's certificate; if Alice's certificate is not on the CRL, then Bob knows that Alice's certificate is still valid.

- The Online Certificate Status Protocol (OCSP) is an Internet protocol used for obtaining the revocation status of a digital certificate. It was created as an alternative to CRLs, specifically addressing certain problems associated with using CRLs in a PKI. Using the example above, Bob would send Alice's certificate serial number in an OCSP request to the CA that issued Alice's certificate. The CA would look up the revocation status of Alice's certificate in its database and the CA's OCSP responder would return a response to Bob indicating whether Alice's certificate is valid.[47]

A PKI operates on three core concepts:

- Policy, defining the controls that must be met by a PKI
- Protocol, defining the means to communicate specific messages
- Format, defining the technical formats for messages

Collectively, under a given policy and using secure protocols, formatted messages can be sent that establish, convey and revoke trust between domains.

6.3.1 Trust Models

PKI uses all cryptography described in Section 6.2 to issue and enable use of certificates.

Both hierarchical (Figure 16) and mesh (Figure 17) trust models can be implemented for PKI, affecting how different entities trust each other's certificates.

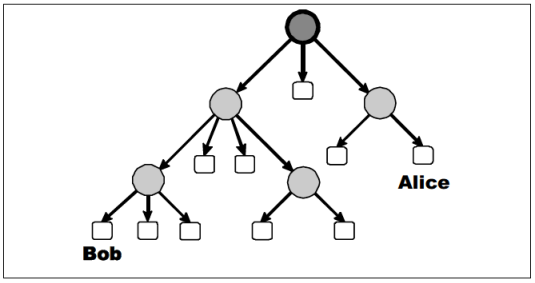


Figure 16. Hierarchical Architecture

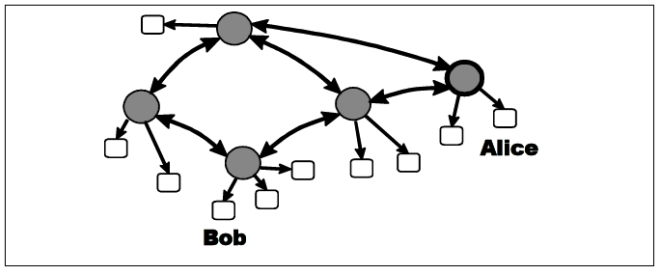


Figure 17. Mesh Architecture

[46] http://en.wikipedia.org/wiki/Revocation_list
[47] http://en.wikipedia.org/wiki/Online_Certificate_Status_Protocol

One example is the U.S. Federal Bridge Certificate Authority (FBCA).[48] The FBCA was established by the U.S. Federal government to extend trust across all federal agencies and is the chief mechanism for enabling trust between industry (external) PKI and Federal (internal) PKI implementations. Figure 18 illustrates the trust architecture for the FBCA.

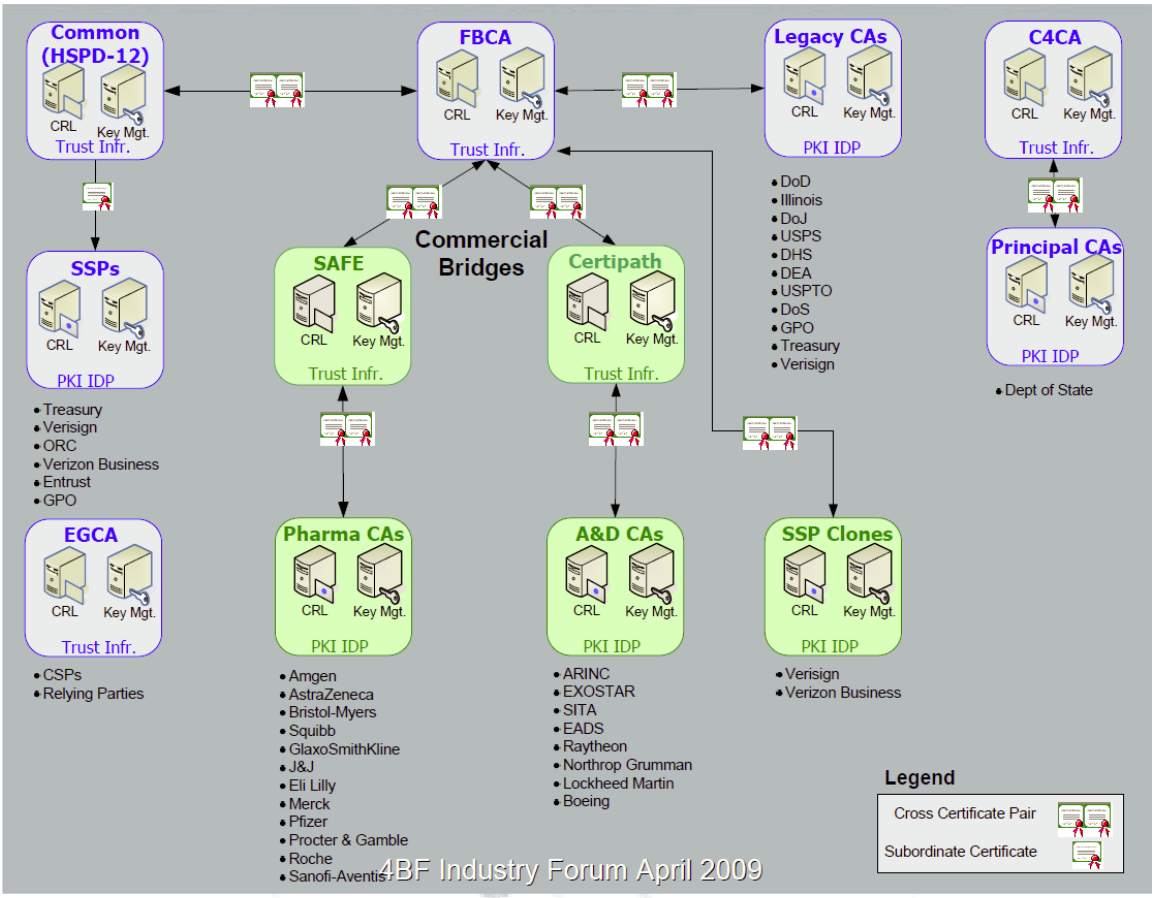


Figure 18. Federal PKI Trust Fabric

6.3.2 Policies

Certificate policies (CP), certificate practices statements (CPS), and key recovery policies (KRP) are defined in a large body of Internet Engineering Task Force (IETF) work. Any certificate CP/CPS/KRP should conform to the IETF RFC 3647[49] for certificate policy and certification practice statement construction.

6.3.2.1 Certificate Policy

The first step in creating a CA is to define a corporate certificate policy (CP)[50]. The CP is the administrative policy that defines all aspects of the certificate lifecycle. A CP addresses all aspects associated with the generation, production, distribution, accounting, compromise recovery, and

[48] For additional information on the Federal Bridge Certificate Authority, see http://www.idmanagement.gov/fpkia/

[49] IETF RFC 3675, *Internet X.509 Public Key Infrastructure, Certificate Policy and Certification Practices Framework*, http://www.ietf.org/rfc/rfc3647.txt

[50] For a robust, fully vetted and cross certified policies, please refer to http://www.certipath.com/policy-docs.htm.

administration of digital certificates. It is strongly recommended that anyone setting up a CA leverage a common baseline policy, separated into a number of assurance levels. An example of a strong baseline is the CertiPath Bridge CA. These policies ensure that everyone is abiding by the same rules and can trust each other at a commensurate level of assurance relative to the sensitivity of the data or situation. The following is text from the CertiPath Bridge CA Certificate Policy:

> This Certificate Policy (CP) defines six certificate policies to facilitate interoperability among Aerospace industry Public Key Infrastructure domains. The six policies represent the medium-software, medium-hardware, medium-CBP-software1, medium-CBP-hardware high-hardware, high-CBP-hardware assurance levels for public key certificates.
>
> The word "assurance" used in this CP means how well a Relying Party can be certain of the identity binding between the public key and the individual whose subject name is cited in the certificate. In addition, it also reflects how well the Relying Party can be certain that the individual whose subject name is cited in the certificate is controlling the use of the private key that corresponds to the public key in the certificate, and how securely the system which was used to produce the certificate and (if appropriate) deliver the private key to the subscriber performs its task.

Another source for an accepted certificate policy is the U.S. Federal PKI Common Policy[51]. These policies assist interoperability among aerospace industry enterprise PKI domains in a peer-to-peer fashion.

6.3.2.2 Certificate Practices Statement

The CPS details the practices the enterprise CA employs in implementing the policies detailed in the enterprise CP. As an example, where a CP might require nightly backups that are moved offsite, the CPS would describe how the backups are done and what software is used for the backup. Further, the provider of the offsite storage would be discussed with details on how the handoff occurs, whether the tapes are in a locked box, and how the chain of custody can be proven.

The main subject area of the standard operating procedures (SOPs) incorporated by reference in the CPS are:

- Securely managing the core infrastructure that supports the CA, and
- Issuing, managing, revoking and renewing the enterprise-issued certificates in accordance with the requirements of the CP

6.3.2.3 Key Recovery Policy (KRP) and Practices Statement (KRPS)

The KRP addresses the security conditions under which key recovery information may be created and to whom and under what security conditions the key recovery information may be released. The KRP also indicates the allowable key recovery agents (KRAs) and how or where key recovery information (KRI) must be maintained.

Encryption keys are a special case since most high value uses of encryption certificates require private keys be escrowed under very tight control. Key escrow is done in the event the end-entity is not available at a later time when the data needs to be recovered. Most PKI implementations will wish to encrypt data, especially for secure email using the S/MIME standards. Leveraging existing examples of KRPs should be evaluated and if feasible, used.

[51] The Federal PKI Common Policy can be found at http://idmanagement.gov.

6.3.3 Deployment Strategies

Setting up a PKI with one or more CAs is not trivial. An organization can choose in-source, out-source or co-source strategy based on the business needs, size of the organization, in-house expertise and economics of the certificates being issued. Three options are summarized in Figure 19 using the CertiPath Bridge CA as an example.

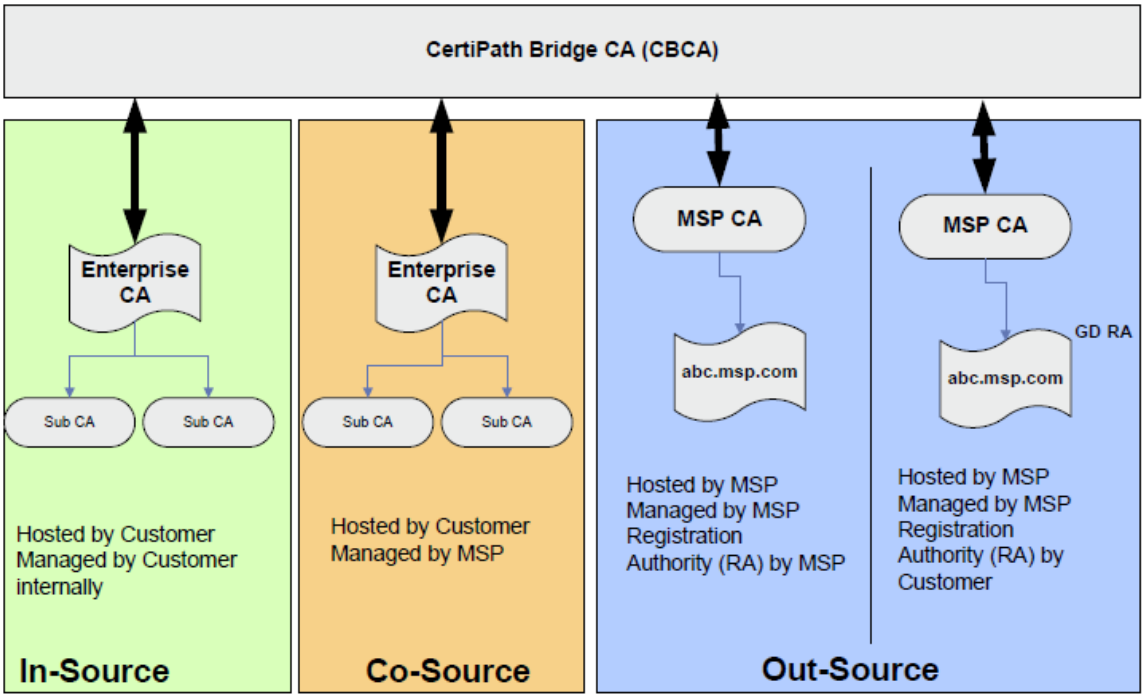


***Figure 19*. PKI Deployment Strategies**

In the in-source model, the CA hosting and management is done internally by the enterprise customer. The business drivers that necessitate deployment and management of an in-house CA are:

- Ability to create subordinate CAs based on business units, geography or other business drivers
- Business, economic or political drivers to maintain root CA
- Ability to define organizational CP
- Flexibility to define the CPS unique to the organization

If these are not business drivers for the enterprise, then the organization can explore two other options.

In the co-source model, the organization outsources the management of the CA but hosts the CA in-house. Examples of co-source providers are VeriSign, Verizon/CyberTrust and Exostar.

In the out-source model, the CA hosting and management is handled by the managed service provider (MSP). The main variation in the out-source model is whether the registration authority (RA) is managed by the MSP or by the customer.

6.4 Smart Cards and Cryptography

As has been discussed throughout this module, keeping the *private key* safe and secure is a significant requirement. For most financial systems, *symmetric keys* must be the focus of protection. Any unauthorized user of a private key or a symmetric key becomes an imposter and is able to act as the authorized user. This can be a very dangerous liability for an application provider. Placing these keys on an easy-to-use, highly portable and secure smart card credentials significantly reduces this liability. The

smart card is capable of being certified against FIPS 140 and Common Criteria requirements, ensuring that the private key and/or symmetric key *stays* private.

This section presents several examples of smart cards being used with cryptography mechanisms discussed in this module.

6.4.1 Identity Credentials

Organizations worldwide are using smart card based identity credentials.

One of the key goals of PKI within an application infrastructure is to enable secure email messaging and digital signatures between parties.

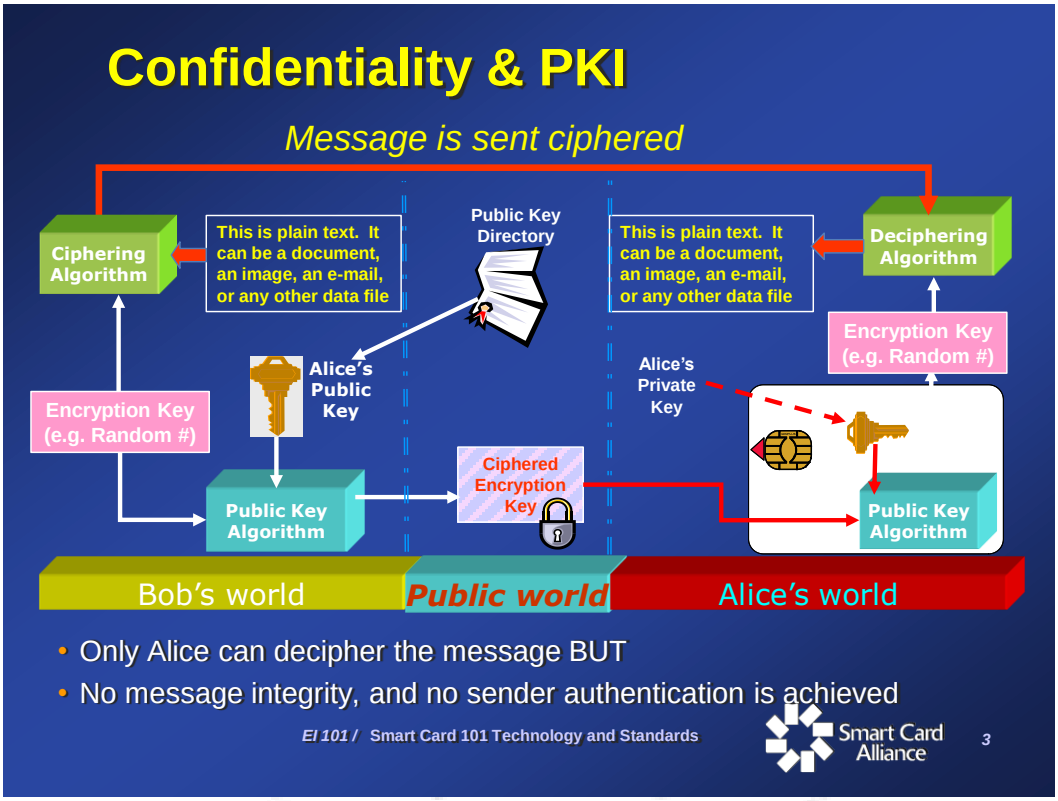


Figure 20. Smart Cards, Confidentiality and PKI

Figure 20 shows a model for a secure email message using a smart card. As seen earlier in this module, PKI and encryption are leveraged to enable a full message to be sent, ciphered and protected, from the originator (Bob) to a specific individual holding the private key (Alice). In the figure above, Bob uses Alice's public key to send a ciphered encryption key to Alice. Bob then uses the encryption key to cipher a message that is sent to Alice. Alice uses the private key stored on her smart card to decipher the encryption key and then decipher the message. Note that the message is never sent in the clear, nor is the key to decrypt the message sent in the clear. The message and key are protected through the use of public key cryptography.

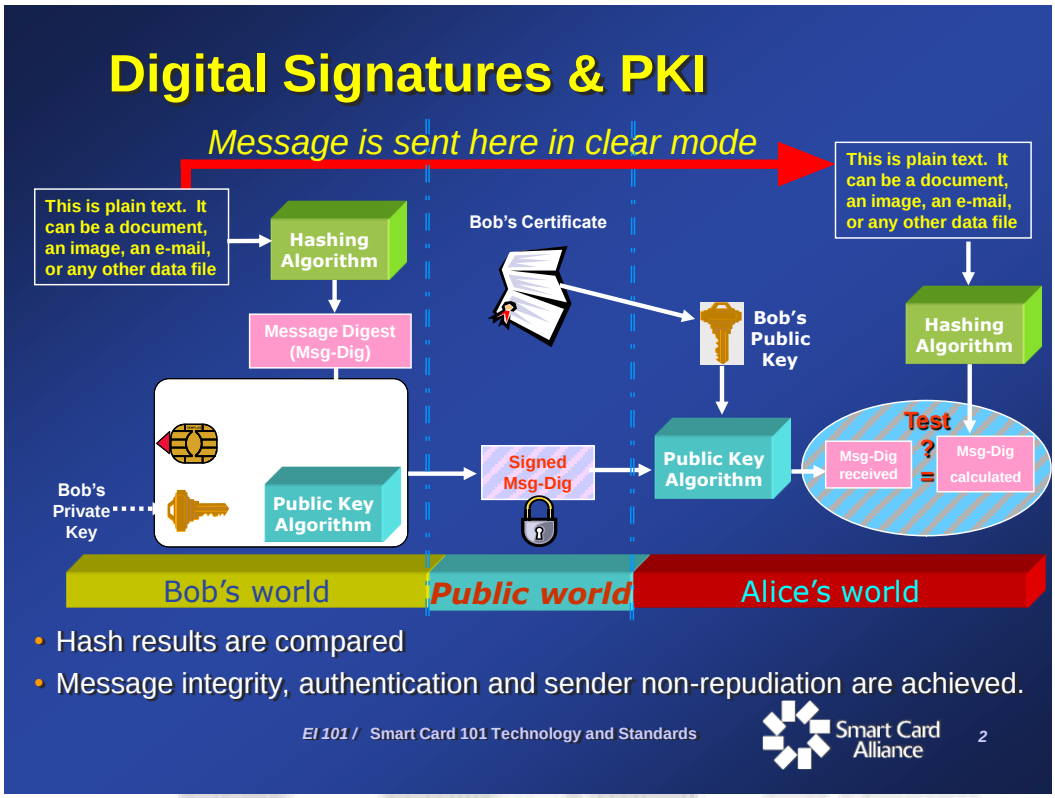


Figure 21. Smart Cards, Digital Signatures and PKI

Figure 21 extends the use of private keys to also enable a digital signature to be generated by Bob, using the private key stored on his smart card. Only Bob could have done this since it is using his smart card that contains his private key. Alice is able to confirm that the message was sent by Bob by receiving Bob's signature (shown by the lock in the figure). Alice uses a directory to acquire Bob's PKI certificate. Contained in that certificate is Bob's public key. Alice then uses this public key to confirm the signature for that document.

6.4.2 Financial

The financial infrastructure uses smart cards and cryptography extensively to protect financial transactions. Two core applications are used in the market today: contact chip payment schemes and contactless chip payment schemes. The EMV specification is the global standard for chip cards used for payment applications.

Figure 22 shows a simplified use of the EMV protocol. Note that the entire process uses symmetric keys to protect the message (a payment instruction) to ensure security of the payment system worldwide. In this model, the personal identification number (PIN) code is used as the cardholder verification method (verifying that the person entering the PIN owns the card) unlock the smart card which enables the cryptographic protocol to run and generate payment instructions to the POS terminal. This application allows both the terminal to authenticate the card and the card to authenticate the terminal.

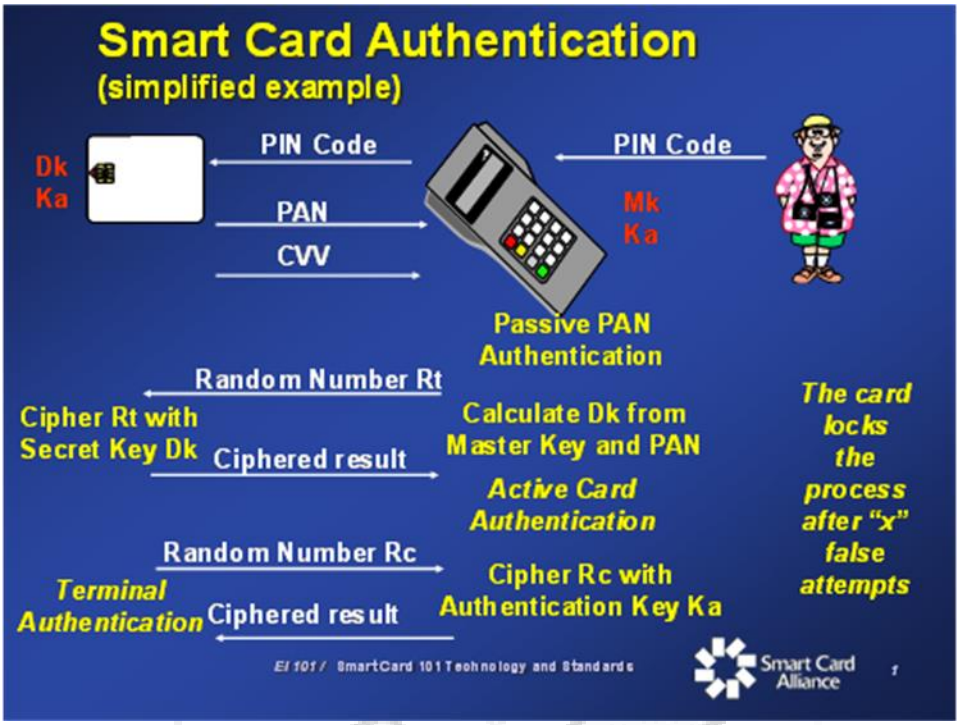


Figure 22. Financial Transaction using Symmetric Cryptography

7 Security at the System Level[52]

The smart card itself is only one component in a smart card-based system implementation. Security mechanisms can be implemented in the card and at the OS, software, and system levels. This section takes a holistic view of security for an application.

No single mechanism provides complete security. Indeed, absolute security is not possible. The objective must be to ensure that the time and effort required to compromise a system yield no gain to the attacker. Risk to the system must be matched with security countermeasures to reduce exposure to a level where the risk can be tolerated. When a smart card-based system is being developed, the countermeasures that will protect the data must incorporate hardware, software, and system security features. This combination of security mechanisms is often referred to as "layers of security."

An issuer's decision on the level of security and type of security measures that should be implemented for an application in a smart card-based system must balance the risks or threats that the issuer expects to encounter, the cost of implementing the security features, and the impact that those features may have on use of the smart card. Not all applications require the same level of security. Application security requirements must be defined when a system is being designed so that the issuer can select the appropriate technology and approach for implementation.

7.1 Security Overview

Security mechanisms are implemented across an entire system. Advances in IC technology allow many security features to be implemented at the IC level. These features are designed to protect the memory contents of the IC and prevent or counter any attacks. Many of the more common features are described in Section 3.2. Additional countermeasures are also available that are proprietary to individual manufacturers and remain confidential.

Some attacks are designed to exploit the physical characteristics of the silicon IC, relying on the limitations of physics. An example of such an attack is a power analysis attack (DPA or SPA). IC manufacturers have implemented many different features to confuse an attacker and prevent critical data from being obtained. However, by adding a layer of security with software in the OS, physical operations can be masked even further, strengthening such countermeasures significantly.

Close cooperation between hardware and software developers can result in additional security layers that can strengthen any given secure IC feature. Software strengthens hardware and vice versa. The end result is a much more secure product.

To design a secure smart card system, designers must look beyond the secure IC itself. In the event that a card is compromised, the system designer must ensure that the security of the whole system is not at risk. For example, accessing the contents of a secure IC's memory should not reveal any master keys that are used throughout the system. Breaking into one card should not provide the opportunity for breaking into many cards.[53] Several mechanisms can be incorporated within the system that will allow additional security enhancements while the cards are deployed in the field. One methodology is to populate the IC's firmware with multiple cryptographic keys. The keys could be changed randomly or at defined intervals based on a number of transactions or a time interval criteria. Another way to update the security of the system is to enable the operating system to accept secure software downloads that would allow for new security features to be added to the IC's software as the need arises. The system could also have a suite of security features that are controlled by the issuer's back office system.

One good example of how to build additional layers of security into such a system can be found in the global payment systems implemented by the financial industry (described in the next section).

[52] *What Makes a Smart Card Secure?*, Smart Card Alliance white paper, October 2008.

[53] This is one of the key benefits of well-designed smart card systems today. The high costs involved in breaking one card must be repeated for every card, making the process financially unviable for the would-be attacker.

7.2 Security and the Financial Payments Industry

The financial payments industry has designed multiple layers of security into the traditional credit and debit payment systems to protect all parties involved in a payment transaction. Most of these protective measures are independent of the technology used to transfer payment account information from the payment card or device to the merchant POS terminal and are used for both magnetic stripe and contactless smart card transactions. For example, online authorization, risk management, and fraud detection systems are used to detect potential fraudulent activity for credit or debit card payment transactions. In addition, the payment brands have liability policies that protect consumers using traditional consumer credit and debit accounts.

7.2.1 Security and Contactless Payments

The financial industry has added security technology to payment systems to prevent fraud for contactless payments; these security measures are implemented both on the contactless device and in the processing network and system.[54] While implementations differ among issuers, examples of security measures that are being used include:

- At the card level, each contactless card can have its own unique built-in secret "key" that uses standard 128-bit encryption technology to generate a unique card verification value (e.g., CVV or CVC) or a dynamic cryptogram that exclusively identifies each transaction. No two cards share the same key, and the key is never transmitted.
- At the system level, payment networks have the ability to automatically detect and reject any attempt to use the same transaction information more than once. Thus, even if someone should "read" information from a contactless transaction or even multiple transactions from the same card, the information would be useless.
- Contactless payment does not require the cardholder's name to be exchanged between the card and the terminal. In fact, best practices within the industry do not include storing the cardholder's name in the contactless IC.
- Some contactless payment cards and devices do not include the cardholder's account number but use an alternate number that is associated with a payment account by the issuer's back-end processing system. This alternate number cannot be used in other payment transactions (e.g., with a magnetic stripe card or over the Internet).

Two types of contactless payments implementation have been implemented, one that originated for the U.S. market that uses magnetic stripe data (MSD) and one based on EMV. They share the security method of having dynamic data in each transaction; however the EMV contactless transaction is capable of supporting all EMV security functionality. (See Section 7.2.2.)

7.2.2 Security and EMV Payments[55]

When discussing smart card security and financial payment, it is appropriate to consider the impact that the introduction of EMV payment cards has had around the world. EMV smart cards have been introduced in Europe, Asia, Latin America and Canada, with the objective of reducing fraud.

The secure microcontroller used in EMV credit and debit cards has allowed the payment industry to implement security features in addition to those available on magnetic stripe cards, such as the following:

- Card authentication, allowing a POS terminal to use cryptography to determine that a card is genuine. Three techniques are used: static data authentication (SDA), dynamic data authentication (DDA), and combined dynamic data authentication/application cryptogram generation (CDA).

[54] Additional information on contactless payments can be found on the Smart Card Alliance web site at http://www.smartcardalliance.org/pages/activities-councils-contactless-payments-resources.

[55] Additional information on EMV specifications can be found on the EMVCo web site, http://www.emvco.com/.

- Cardholder verification, allowing the cardholder to use a personal identification number (PIN) (if not compromised) to confirm that the valid cardholder is present.

The features of the secure microcontroller in an EMV card can also enhance control over transaction authorization based on the cardholder's spending behavior. The microcontroller can support offline transactions and decide itself whether to go online, instead of having to rely strictly on merchant floor limits. In addition:

- The IC can be locked against future use by an online message from the issuer.
- 1-in-N counters can decide how many transactions can occur without online authorization. This feature can limit the number of transactions occurring below the cardholder's credit limit.
- A value counter can track the cumulative amount spent between online authorizations, trigger an online authorization, and return control to the issuer.

If the secure microcontroller in the EMV card is used to verify both the cardholder and the card itself, most "face to face" fraud can be eliminated, including fraud related to lost/stolen cards, cards never received by mail, and counterfeit cards. The card can be authenticated by checking that it has not been altered using an issuer-programmed algorithm encrypted in the IC. This process can be carried out offline between the card and the terminal.

The identity of the cardholder can also be validated by requiring the use of a PIN or other methods as set forth in the EMV specification. For PIN validation, the process checks that the PIN entered by the cardholder matches the PIN encrypted on EMV card's secure IC. Counters can prevent repeated attempts to guess a PIN and block use of the card.

By using a secure microcontroller, EMV smart cards prevent fraud caused by criminals skimming cardholder information from the credit or debit card's magnetic stripe or by embossing numbers on counterfeit cards.

The availability of transaction certificates and digital signatures with EMV payment cards can reduce merchant fraud through the use of cryptography (as described above) for non-repudiation and certified transactions.

EMV also allows issuers to use scripts to modify data elements (such as the PIN or risk parameters) on an EMV smart card during online transactions.

To support online transactions, issuers are required to receive extra IC-related data in the online message and reply to the acquirer, and therefore to the device, with additional response data. This includes authentication using the authorization request cryptogram (ARQC) and authorization response cryptogram (ARPC) in a process known as online mutual authentication (OMA).

7.3 Transit[56]

Transit agencies worldwide have implemented automatic fare collection (AFC) systems that use contactless smart card technology for transit-issued fare media. These systems are popular since they deliver fast, easy access to riders and reduced operating costs and improved efficiencies to transit operators. Like the financial industry, the transit industry employs layers of security throughout the fare payment system.

- Contactless transit fare payment cards use cryptographic techniques to protect data stored on the card and to authenticate the card to the fare collection system. While historically the transit industry used proprietary approaches and short encryption keys, new systems are using ISO/IEC 14443-based contactless cards with stronger cryptography.

[56] *Transit Payment System Security*, Smart Card Alliance white paper, August 2008.

- Fraud prevention in transit payment systems is typically based on making it very difficult to counterfeit the contactless smart cards that are used to pay fares, to trick the system into thinking a different card is a legitimate fare card, or to add value to a legitimate card without paying. Preventive measures include card security (e.g., using diversified keys), reader security and system-level functions (e.g., hot lists, transaction monitoring).
- Detection is a critical element of any security system. Knowing when a counterfeit card has been successfully used is essential to confining losses to small amounts. All transit systems include transaction audits and reporting measures that enable security features to detect improper use.
- Maintaining the integrity of and managing access to the transit payment system are critically important. Transit agencies manage this risk through appropriate physical access control to facilities and logical access control to networks and computers.
- From the consumer's point of view, transit smart cards are very secure, as most systems feature rider anonymity on the card. A few exceptions exist, for example, with people who qualify for special fares. In this case, the consumer uses a registration process to personalize the card with a picture and electronic information specific to the entitlement and the individual. Even in this case, the card, if registered, usually contains a unique number that can then be used only by the back office to determine the owner.
- Customer data is typically held at the back office and is not stored on the card itself. For the most part, even when cards are registered for balance protection or autoload features, transit systems manage personal data at the central system and follow published security system standards to design necessary protection.

7.4 Mobile: SIM and UICC[57]

As early as 1982, radio telephony was being standardized at an industry level through a set of highly detailed specifications across Europe under the auspices of Conférence Européenes des Postes et Télécommunications (CEPT). The industry level standardization continued under the European Telecommunications Standards Institute (ETSI).

The mobile network was named Groupe Spécial Mobile (GSM) but later the meaning of the acronym was changed to “Global System for Mobile telecommunications.”

Smart cards were part of the GSM standard from almost the beginning. The role of the smart card in GSM is primarily to hold subscriber information. The smart card used in GSM is called a Subscriber Identity Module (SIM). The GSM industry specification for the SIM is the *Specification of the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface (GSM 11.11)*. Terminology in GSM 11.11 (and elsewhere in ETSI) is very specific to the mobile industry and it is generally a good idea to become familiar with the technical vocabulary before diving too far into the specifics.

A mobile phone, as defined in GSM, is known as a mobile station (MS). An MS (i.e., the handset) itself consists of two primary components, the mobile equipment (ME) and the SIM.

The GSM transmission system consists of a network operator which supports a variety of mobile switching equipment, operations center, helpdesk, and billing platforms and that is connected to several transmission/reception stations each known as a base station (BS). A BS in turn is composed of a base station controller (BSC) and an antenna tower.

---

[57] Sources: Smart Card Handbook, by Wolfgang Rankl and Wolfgang Effing, Wiley, 2004; Smart Cards: The Developer’s Toolkit, by Timothy M. Jurgensen and Scott B. Guthery, Prentice Hall, 2002; Mobile Application Development with SMS and the SIM Toolkit, by Scott Guthery and Mary Cronin, McGraw Hill Professional, 2001; European Telecommunications Standards Institute (ETSI), http://www.etsi.org; Neville Pattinson, Gemalto

Two critical sub-systems in GSM are the authentication center (to authenticate subscribers when each attempts to connect to the network) and the mobile switching center (to provide voice and data services to an authenticated subscriber).

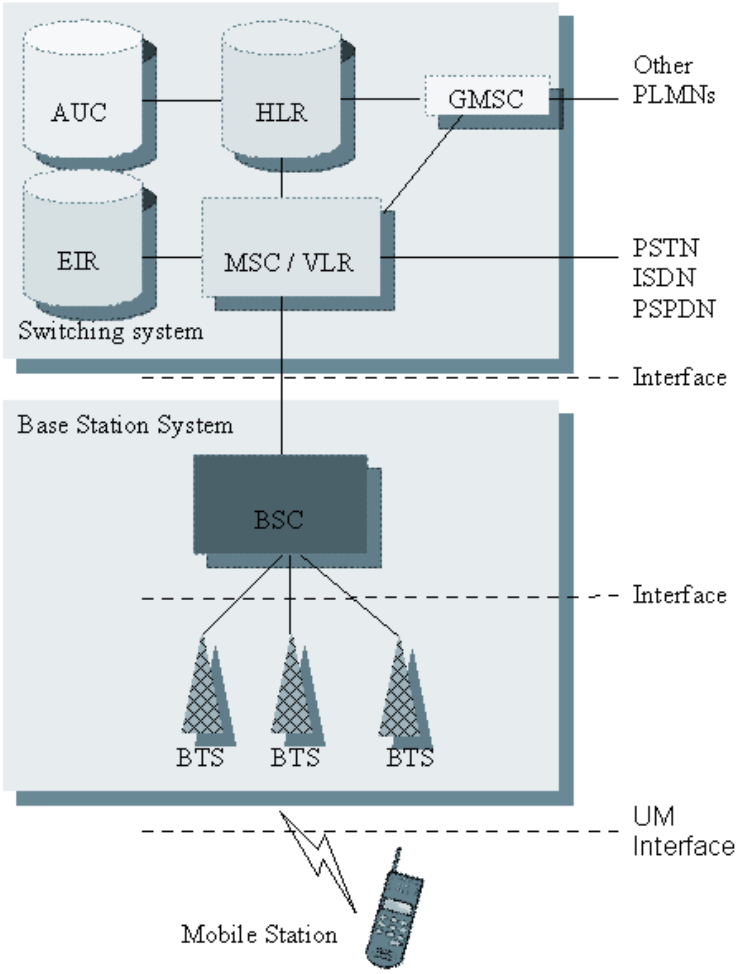


Figure 23. Major Components in the GSM Network[58]

The authentication requirement helped to ensure a permanent role for the smart card. The SIM holds a secret key (named the Ki secret key) along with other subscriber account information that is used to authenticate the MS to the network.

Several benefits resulted in putting this authentication function in the SIM instead of the MS. The SIM can be owned and issued to the subscriber independent of the type of handset to which it is associated. The SIM is portable, can be used in multiple handsets and is useful for:

1. A subscriber upgrading to a new phone within the network
2. A subscriber renting a phone when roaming on a foreign network
3. A subscriber using a loaner phone while a phone is being repaired

The original scope of the SIM card was as a medium for user identification (via a personal identification number (PIN)), authentication to the network, and secure storage for subscriber information that is needed for billing purposes in a manner independent of the handset.

[58] Source: http://www.ipv6.com/articles/mobile/GSM-Mobile-Networks.htm

The GSM network allows activation of a handset using a technology known as over-the-air (OTA) communications. This is accomplished by using the Short Message Services (SMS) that GSM enabled long before "texting" was popular in other parts of the world. Short messages are sent to the SIM where they are later retrieved by the handset under subscriber control. OTA creates a unique type of short message that provides instructions (and requested responses) to and from the SIM. This allows remote personalization of the SIM to be achieved through a GSM network.

The ability to remotely activate a SIM was enhanced by the 1996 publication of the SIM Application Toolkit specification. This specification was successful because it built upon existing GSM network services without significant changes to both the network and the handset.

The SIM Applications Toolkit allows an application to be downloaded to the handset using the SIM as receiver of the application. Once successfully downloaded, the application could be executed in a manner that allowed the SIM to control aspects of the phone including the display, key entry, and the ability to originate application-specific short messages.

SIM cards today support large memory, have higher transfer speeds, and work at very low voltage to conserve power.

More recently, mobile network operators are facing huge growth in data traffic due to an every-increasing level of mobile Internet usage. The growth is largely driven by a growing consumer love affair with smart phones, although USB modems and other mobile Internet devices contribute as well. These devices offer end users a wide selection of rich Internet-based services including messaging, localization, interaction with social communities, publication and consumption of content, and more. As a result, major operators worldwide are planning to migrate to high bandwidth all-IP wireless networks capable of supporting this data traffic and, in the medium term, allowing IP Multimedia Subsystem (IMS)-based voice and SMS.

The migration to Long Term Evolution (LTE) is underway by many mobile operators around the world. The LTE standard is developed and maintained by the 3rd Generation Partnership Project (3GPP) standards body. The LTE network represents an advance on existing Universal Mobile Telecommunications System (UMTS or 3rd Generation, 3G) networks, and is also the commonly accepted evolution path for all currently deployed GSM, W-CDMA and CDMA networks. These LTE networks represent a new era of connectivity for consumers and offer new opportunities for mobile operators and the mobile ecosystem. LTE takes advantage of an enhanced radio interface and the scalability of Internet architectures and combines these with the security of the Universal Integrated Circuit Card (UICC) and associated network-based solutions and services.

7.4.1 The Anatomy of a SIM

A SIM card has data organized in a manner compatible with ISO/IEC 7816.

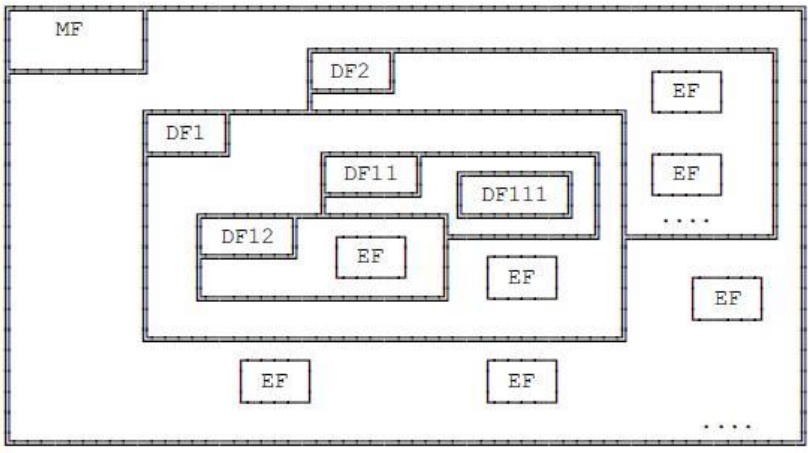


Figure 24. SIM File Hierarchy (Source ETSI GSM 11.11)

Each data file, known as an Elementary File (EF) in the ISO standard, is identified by two hexadecimal bytes. For example, the EF that holds the last number dialed is accessed by selecting this file using the File Identifier (FID) of '6F 44' and then reading the most recent fixed size record contained in this EF.

The hierarchy of a GSM card is composed of multiple levels of "application" files known as Dedicated Files (DF), each with either several EF structures or additional DF structures within a DF context. This hierarchy continues to evolve, albeit in a standard way under the control of ETSI. Such specifications are mandatory to ensure global interoperability of the GSM solution.

7.4.2 LTE and the UICC

The UICC is the smart card used in mobile phones and terminals for GSM and UMTS/3G networks. It authenticates the subscriber to the network just as the SIM's role in GSM, while ensuring the integrity and security of the user's personal data. The UICC also stores applications for both operator and end-user use for the correct deployment of mobile services. The UICC brings a whole host of fundamental features, tried and tested within GSM and now perfected for LTE. Fully integrated into IP networks, the UICC, as per existing standards, has become an IP-connected processor with its own IP layers and IP stack. This means that it is ready for deployment in all IP networks such as LTE. The UICC can be supplied in different form factors to suit various business models – standard USIM card, USB dongle, SIM in 3G laptops and smart modems. The UICC can store several gigabytes of personal content for the user.

8 References

3GPP, http://www.3gpp.org

Application of Attack Potential to Smartcard, version 2.5, Joint Interpretation Library, November 2007, http://www.ssi.gouv.fr/site_documents/JIL/JIL-Application-of-Attack-Potential-to-Smartcards-V2-5.pdf

Applied Cryptography: Protocols, Algorithms and Source Code in C, by Bruce Schneier, Wiley 1996

CertiPath web site, http://www.certipath.com

Common Criteria, http://www.commoncriteriaportal.org/

Contactless Technology Security Issues, Smart Card Security, Helena Handschuh, Information Security Bulletin, Volume 9, April 2004.

EMVCO web site, http://www.emvco.com

European Telecommunications Standards Institute (ETSI) web site, http://www.etsi.org

Eurosmart web site, http://www.eurosmart.com

Federal Bridge Certificate Authority, http://www.idmanagement.gov/fpkia/

FIPS 180-3, *Secure Hash Standards (SHS),* October 2008

FIPS 197, *Advanced Encryption Standard*, November 2001

Federal Information Processing Standard (FIPS) 201 Personal Identity Verification (PIV) of Federal Employees and Contractors, http://csrc.nist.gov/publications/fips/fips201-1/FIPS-201-1-chng1.pdf

Godzilla Crypto Tutorial, by Peter Gutmann, http://www.cs.auckland.ac.nz/~pgut001/tutorial/index.html

IETF RFC 3675, *Internet X.509 Public Key Infrastructure, Certificate Policy and Certification Practices Framework*, http://www.ietf.org/rfc/rfc3647.txt

Initiative for Open Authentication (OATH), http://www.openauthentication.org

ISO/IEC, http://www.iso.org

Mobile Application Development with SMS and the SIM Toolkit, by Scott Guthery and Mary Cronin (McGraw Hill Professional, 2001)

NIST FIPS publications, http://csrc.nist.gov/publications/PubsFIPS.html

NIST PIV web site, http://csrc.nist.gov/groups/SNS/piv/index.html

NIST SP800-21-1, *Guideline for Implementing Cryptography in the Federal Government, Second Edition*, December 2005

NIST SP800-57, *Recommendation for Key Management, Parts 1 and 2*, August, 2005

NIST SP 800-67, *Recommendation for the Triple Data Encryption Algorithm (TDEA) Block Cipher*, May 2008

NIST SP 800-78, *Cryptographic Algorithms and Key Sizes for Personal Identity Verification*, August 2007

Smart Card Handbook, by Wolfgang Rankl and Wolfgang Effing (Wiley, 2010), http://www.wrankl.de/SCH/SCH.html

Smart Cards: The Developer's Toolkit, by Timothy M. Jurgensen and Scott B. Guthery (Prentice Hall, 2002)

Transit Payment System Security, Smart Card Alliance white paper, August 2008, http://www.smartcardalliance.org

Understanding Secure Contactless Device versus RFID Tag, Eurosmart, http://www.eurosmart.com/4-Documents/PositionPapers.htm

What Makes a Smart Card Secure?, Smart Card Alliance white paper, October 2008, http://www.smartcardalliance.org

9 Acknowledgements

This document was developed by the Smart Card Alliance for the Certified Smart Card Industry Professional (CSCIP) program. Publication of this document by the Smart Card Alliance does not imply the endorsement of any of the member organizations of the Alliance.

The Smart Card Alliance thanks the following individuals and organizations for their review of this CSCIP module:

- **Christophe Goyet, Oberthur Technologies**
- **Bryan Ichikawa, Unisys**
- **Gilles Lisimaque, Identification Technology Partners**
- **Neville Pattinson, Gemalto**

The Smart Card Alliance thanks the following individuals and organizations for contributing content to this CSCIP module:

- **Gilles Lisimaque, Identification Technology Partners**, Section 4, *Card Edge Interface Security*
- **Steve Howard, CertiPath LLC**, Section 6, *Cryptography and Public Key Infrastructure*
- **Neville Pattinson, Gemalto**, Section 7.4, *Mobile: SIM and UICC*
- **Gerald Smith, Identification Technology Partners**, Section 7.4, *Mobile: SIM* and UICC

The Smart Card Alliance thanks the many current and past members of the Smart Card Alliance Councils and Task Forces who contributed to the development of the white papers and reference material that was used to create this module.

About LEAP and the CSCIP Program

The Smart Card Alliance Leadership, Education and Advancement Program (LEAP) was formed to: offer a new individual members-only organization for smart card professional; advance education and professional development for individuals working in the smart card industry; manage and confer, based on a standardized body-of-knowledge examination, the Certified Smart Card Industry Professional (CSCIP) designation.

LEAP members who wish to achieve certification as experts in smart card technology may do so at any time. Certification requires that LEAP members meet specific educational and professional criteria prior to acceptance into the certification program.

A series of educational modules forming the CSCIP certification body of knowledge has been developed by leading smart card industry professionals and is updated regularly. These educational modules prepare applicants for the multi-part CSCIP exam administered by the Smart Card Alliance. The exam requires demonstrated proficiency in a broad body of industry knowledge, as opposed to expertise in specialized smart card disciplines. Applicants must receive a passing grade on all parts of the exam to receive the CSCIP certification.

LEAP membership in good standing is required to sustain the certification, and documentation of a required level of continuing education activities must be submitted every three years for CSCIP re-certification.

Additional information on LEAP and the CSCIP accreditation program can be found at http://www.smartcardalliance.org.

Trademark Notice