National Information Assurance Partnership

Common Criteria Evaluation and Validation Scheme
Validation Report

United States Marine Corps Public Key Infrastructure Framework (PKIF) Version 1.2

Report Number:	**CCEVS-VR-06-0022**
Dated:	**8 August 2006**
Version:	**1.0**

National Institute of Standards and Technology
Information Technology Laboratory
100 Bureau Drive
Gaithersburg, MD 20899

National Security Agency
Information Assurance Directorate
9800 Savage Road STE 6740
Fort George G. Meade, MD 20755-6740

USMC Public Key Infrastructure Framework (PKIF) Version 1.2

CCEVS-VR-06-0022

ACKNOWLEDGEMENTS

Validation Team

Richard Murphy
Mitretek Systems
Falls Church, Virginia

Common Criteria Testing Laboratory

Elise Berger
Debra Baker
Gary Grainger
Swapna Katikaneni
Peter Kukura
Jean Petty
CygnaCom Solutions
McLean, Virginia

Table of Contents

1 Executive Summary

This report documents the National Information Assurance Partnership (NIAP) Validator's assessment of the evaluation of the Public Key Infrastructure Framework (PKIF) Version 1.2. It presents the evaluation results, their justifications, and the conformance results. This Validation Report is not an endorsement of the Target of Evaluation (TOE) by any agency of the U.S. Government and no warranty of the TOE is either expressed or implied.

The evaluation of the PKIF was performed by CygnaCom Solutions Common Criteria Testing Laboratory (CCTL) in the United States and was completed during April 2006. The information in this report is largely derived from the Evaluation Technical Report (ETR) and associated test report, both written by CygnaCom.

The evaluation was carried out in accordance to the Common Criteria Evaluation and Validation Scheme (CCEVS) processes and procedures. PKIF Version 1.2 was evaluated against the criteria contained in the Common Criteria for Information Technology Security Evaluation, Version 2.2. The evaluation methodology used by the evaluation team to conduct the evaluation is the Common Methodology for Information Technology Security Evaluation, Version 2.2 (CEM).

CygnaCom Solutions determined that the product meets the security criteria in the Security Target, which specifies an assurance level of Evaluation Assurance Level (EAL) 4 augmented with ALC_FLR.1 (Basic Flaw Remediation). The evaluation team determined the product to be Part 2 extended and Part 3 conformant, and concluded that the Common Criteria requirements for EAL 4 augmented have been met. The evaluation team also determined that the TOE is conformant with a Protection Profile PP selected from the *U.S. Government Family of Protection Profiles for Public Key-Enabled Applications, Version: 2.61,* July 31, 2004 [PP]. The security functional requirements (SFRs) for PPs in this family are derived from 15 SFR packages and the base SFRs that are common to all packages. The base SFRs must be satisfied by either the TOE or its environment (see [PP] Section 5.1). The product's PP conformance claim is based on the inclusion of the SFRs from 10 of the 15 packages and an environment that satisfies all base SFRs. The PP's full name, which lists the 10 packages, is given in the Identification section below.

The TOE is a C++ software library designed to simplify the task of adding Public Key Infrastructure (PKI) support to applications. The PKIF library can be used by developers using the Microsoft Visual C++ .NET 2002 Integrated Development Environment. The PKIF relies upon the IT environment for basic cryptographic functions, using the Microsoft Cryptographic Application Programming Interface (CAPI) as the interface to the cryptographic modules provided by the IT environment.

The validation team monitored the activities of the evaluation team, participated in team meetings, provided guidance on technical issues and evaluation processes, reviewed successive versions of the Security Target, reviewed selected evaluation evidence, reviewed test plans, reviewed intermediate evaluation results (i.e., the CEM work units), and reviewed successive versions of the ETR and test report. The validation team determined that the evaluation team showed that the product satisfies all of the functional and assurance requirements defined in the Security Target for an EAL4 evaluation. Therefore the validation team concludes that the CygnaCom findings are accurate, and the conclusions justified.

2 Identification

TOE: Public Key Infrastructure Framework Version 1.2

Evaluated Software: Public Key Infrastructure Framework Version 1.2

Sponsor: United States Marine Corps (USMC)

Protection Profile: U.S. Government PKE PP with
Certification Path Validation (CPV) – Basic Package,
CPV – Basic Policy Package,
CPV – Policy Mapping Package,
CPV – Name Constraints Package,
PKI Signature Generation Package,
PKI Signature Verification Package,
PKI Encryption using Key Transfer Algorithms Package,
PKI Decryption using Key Transfer Algorithms Package,
Online Certificate Status Protocol Client Package, and
Certificate Revocation List (CRL) Validation Package
at EAL4 with augmentation, Version 2.61, July 31, 2004.

CCTL: CygnaCom Solutions
7925 Jones Branch Drive, Suite 5200
McLean, VA 22102

Validation Team: Richard Murphy, Mitretek Systems, Inc.

CC Identification: *Common Criteria for Information Technology Security Evaluation*, Version 2.2, January 2004 [CCV2.2].

CEM Identification: *Common Methodology for Information Technology Security Evaluation*, Version 2.2, Evaluation Methodology, January 2004 [CEMV2.2].

Interpretations: All CCIMB interpretations as of the date of the Kick-off meeting held on March 4, 2004, were considered during the evaluation. No

National or International Interpretations were found to apply to the evaluation.

3 Security Policy

PKIF is a toolkit used by application developers to incorporate secure PKI functionality into an application. As such, PKIF does not enforce a security policy as PKIF has only one role, that being defined as a *user*. The user is considered to be the application using PKIF, or, to provide a human definition, the application developer. There are no untrusted users for the TOE as untrusted users would be interacting with a public key enabled (PKE) application, which would then interface with PKIF on the user's behalf. The developer of the application is trusted to use PKIF properly in accordance with the guidance provided, thus there is no untrusted user threat.

The TOE enforces the following security policies in conjunction with the IT environment

- **Certification Path Processing Policy:** Performs certification path development, certification path validation and revocation status checking. Validates certificates starting with a Trust Anchor ending with the subscriber's certificate, validating that the certificates are valid and conform to usage constraints.
- **Decryption Policy:** Performs private key decryption as defined by the Cryptographic Message Syntax (CMS) standard [RFC_3369]. Decryption is performed either with a private key provided by the application or, if no private key is provided, a decryption key in the user's credential store.
- **Digital Signature Generation Policy:** Generate message signatures as defined by the CMS standard using an application-supplied key and CMS SignedData format. Signers are identified by subject key identifier. Additional attributes indicate what information is included with a signature: certificates, revocation information, and/or attributes.
- **Digital Signature Verification Policy:** Verify message signatures for CMS messages. Verification includes specific attributes and certification path development and validation. If present, the following attributes are checked during signature verification: key usage extension, ContentType, MessageDigest, and timestamp.
- **Encryption Policy:** Performs private key message encryption as defined by CMS. PKIF establishes trust in a recipient's public key for encryption. It establishes trust using certification path development and validation, including a check of the key usage extension, if present. Encryption is performed using the public key and public key algorithm from the recipient's certificate. The encrypted CMS object contains key encryption algorithm, data encryption algorithm, and decryption key identifier.

The security functional requirements for the TOE and the IT environment are documented in section 5 of the ST. A summary of the SFRs for the TOE and IT environment are included in the tables below.

TOE Security Functional Requirements

Class FDP: User Data Protection	
FDP_CPD.1	Certification path development
FDP_DAU_CPV_INI.1	Certification path initialisation -- basic
FDP_DAU_CPV_CER.1	Certificate processing -- basic
FDP_DAU_CPV_CER.2	Intermediate certificate processing -- basic
FDP_DAU_CPV_OUT.1	Certification path output -- basic
FDP_DAU_CPV_INI.2	Certification path initialisation – basic policy
FDP_DAU_CPV_OUT.2	Certification path output – basic policy
FDP_DAU_CPV_INI.3	Certification path initialisation – policy mapping
FDP_DAU_CPV_CER.3	Intermediate certificate processing – policy mapping
FDP_DAU_CPV_OUT.3	Certification path output – policy mapping
FDP_DAU_CPV_INI.4	Certification path initialisation – names
FDP_DAU_CPV_CER.4	Certificate processing – name constraints
FDP_DAU_CPV_CER.5	Intermediate Certificate processing – name constraints
FDP_ETC_SIG.1	Export of PKI Signature
FDP_ITC_SIG.1	Import of PKI Signature
FDP_DAU_SIG.1	Signature Blob Verification
FDP_ETC_ENC.1	Export of PKI Encryption – Key Transfer Algorithms
FDP_DAU_ENC.1	PKI Encryption Verification – Key Transfer
FDP_ITC_ENC.1	Import of PKI Encryption – Key Transfer Algorithms
FDP_DAU_OCS.1	Basic OCSP Client
FDP_DAU_CRL.1	Basic CRL Checking

IT Environment Security Functional Requirements

Class FCS: Cryptographic Support	
FCS_CRM_FPS.1	FIPS compliant cryptographic module
Class FDP: User Data Protection	
FDP_ACC.1	Subset Access Control – PKI Credential Management
FDP_ACF.1	Security attribute based access control – PKI Credential Management
FDP_ITC_PKI_INF.1	Import of PKI information from outside the TSF
Class FIA: Identification and Authentication	
FIA_AFL.1	Authentication Failure Handling
FIA_ATD.1	User attribute definition
FIA_UAU.1	Timing of authentication
FIA_UAU.7	Protected authentication feedback
FIA_UID.1	Timing of identification
Class FMT: Security Management	

FMT_MSA.1	Management of security attributes
FMT_MSA.3	Static attribute initialization
FMT_MTD.1	Management of TSF data
FMT_SMF.1	Specification of management functions
FMT_SMR.2	Restrictions on security roles
Class FPT: Protection of the TSF	
FPT_RVM.1	Non-bypassability of the TSP
FPT_SEP.1	TSF domain separation
FPT_STM.1	Reliable time stamps

4 Assumptions and Clarification of Scope

4.1 Usage Assumptions

For secure usage, the operational environment must be managed in accordance with the documentation associated with the following EAL4 assurance requirements:

ADO_DEL.2	Detection of modification
ADO_IGS.1	Installation, generation, and start-up procedures
AGD_ADM.1	Administrator guidance
AGD_USR.1	User guidance

4.2 Environmental Assumptions

The environmental assumptions listed in the following table are required to ensure the security of the TOE.

Environmental Assumptions

Assumption Name	Description
AE.Authorized_Users	Authorized users are trusted to perform their assigned functions.
AE.Configuration	The TOE will be properly installed and configured.
AE.Crypto_Module	The TOE environment is assumed to include one or more cryptographic module(s) that are all validated at FIPS 140 series Level 1 or higher. This FIPS 140 series validated module or modules will perform one or more of the following: key pair generation, digital signature generation and verification, encryption, decryption, secure hash, random number generation, Hash based Message Authentication Code (HMAC) and/or other required cryptographic functions. In summary, all cryptographic modules in the TOE shall be validated at FIPS 140 series Level 1.
AE.Low	The attack potential on the TOE is assumed to be low.
AE.Physical_Protection	Physical protection is assumed to be provided by the environment. The TOE hardware and software is assumed to be protected from unauthorized physical access.
AE.PKI_Info	The certificate and certificate revocation status information is available to the TOE for the time of interest (TOI).

Assumption Name	Description
AE.Time	Accurate system time with required precision in GMT format is assumed to be provided by the environment.

4.3 Clarification of Scope

The PKIF library has a large number of interfaces (about 1600) that could potentially be manipulated by an attacker. The assumption that the application developers are trusted to properly use the library is used to restrict the external interfaces that comprise the TOE Security Functions (TSF) interface to about 113 methods and functions. This restricted TSFI provides the intended capabilities to application developers while keeping the number of interfaces to a manageable level. All of the tested external interfaces are documented in the PKI Framework Users Guide [PKIFUG].

The security features of PKIF are identified and described in general terms. The TOE provides the following logical scope for security functionality testing: Roles, User Data, TSF Data and Security Audit. Because PKIF is a software library, User Identification and Authentication, Access Control and security management functions are functions of the TOE environment. In this case, the TOE is identical to the product, thus no distinction needs to be made between the product and the parts of the product that comprise the TOE.

5 Architectural Information

PKIF is a C++ software library designed to simplify the task of adding PKI support to applications. PKIF provides application developers a set of extensible classes, packaged as a Windows dynamic link library (DLL), that perform a variety of PKI-related functions including:

- Certification Path Processing
- CMS based Signature Generation
- Verification of signatures on CMS messages using PKI
- CMS based PKI Encryption using Key Transfer Algorithms functionality
- CMS based PKI Decryption using Key Transfer Algorithms functionality
- Online Certificate Status Protocol Client functionality
- Certificate revocation list processing functionality
- ASN.1 encoding/decoding functionality
- Cryptographic message creation and processing in CMS format.

Note, for base cryptographic functions, cryptographic key lengths supported by PKIF are not a function of the PKIF DLL, but rather, are determined by the capabilities of the relevant Cryptographic Service Provider (CSP). For all of the functions provided by PKIF, base cryptographic operations such as hash validation, encryption, and decryption are performed by CSP modules supplied by the IT environment.

5.1 Certification Path Processing

PKIF performs X.509 certification path processing, including certification path development and certification path validation. Certification path validation consists of validating certificates starting with the one certified by a trust anchor and ending with the one issued to the subscriber of interest. PKIF supports X.509 version 3 Certificates and X.509 CRLs, versions 1 and 2. Certification path processing is X.509 and PKIX RFC3280 compliant.

There are three types of public key certificates involved in certificate path validation:

- <u>Trust anchor (TA) certificates</u>: These are certificates containing public keys that do not require any validation. Trust anchors generally take the form of a self-signed certificate. TAs must be delivered to entities that rely on the TA’s public key using trusted means. The primary purpose of the trust anchor is to provide a means of conveying a Distinguished Name (DN), public key, algorithm identifier, and the public key parameters (if applicable) for use in validating certification paths.
- <u>Intermediate certificates</u>: These are the certificates issued to Certificate Authorities (CAs). All certificates in a certification path are intermediate certificates, except the trust anchor certificate and end entity certificate.
- <u>End certificates</u>: This is the last certificate in the certification path and is issued to the subscriber of interest. A subscriber certificate is also called an end-entity certificate (i.e., a certificate issued to an entity not functioning as a CA). Sometimes, the last certificate can be a CA certificate, e.g., when the certification path is used to verify signature on a CRL.

During the CC evaluation, the evaluator verified that PKIF processes the following extensions: ocsp-nocheck, keyUsage, extendedKeyUsage, basicConstraints, certificatePolicies, policyMapping, inhibitAnyPolicy, policyConstraints, and nameConstraints, subjectKeyIdentifier, subjectAltName and crlDistributionPoints.

By default, PKIF assumes that the path validation is being done as of the current system time, as opposed to verification of signature relative to a point in time in the past. However, applications can specify a time other than the current time for use during path validation.

5.2 Signature Generation Functionality

PKIF enables applications to use a private key for signature generation and to specify information covered by that signature and packaged with the signature, e.g. using the CMS SignedData format.

5.3 PKI Signature Verification Functionality

PKIF enables applications to process signature information, e.g. using the CMS SignedData format, and to verify signatures using a public key.

5.4 PKI Encryption using Key Transfer Algorithms Functionality

PKIF enables applications to perform public key encryption using key transfer algorithms such as RSA using CMS EnvelopedData format.

5.5 PKI Decryption using Key Transfer Algorithms Functionality

PKIF enables applications to perform private key decryption of CMS EnvelopedData format using key transfer algorithms such as RSA.

5.6 Online Certificate Status Protocol Client Functionality

PKIF can generate Online Certificate Status Protocol (OSCP) requests and validate OCSP responses to determine the revocation status of public key certificates. PKIF verifies OCSP Responder as a trust anchor, as a CA, or as an end entity authorized to sign OCSP responses. PKIF establishes trust in the OCSP responder certificates by performing Certification Path Validation.

5.7 Certificate Revocation List functionality

PKIF provides Certificate Revocation List (CRL) validation functionality that enables applications to determine the revocation status of a certificate using a CRL. PKIF may be used to process CRLs obtained from a variety of sources including: locations indicated by a CRL Distribution Point (CRLDP) extension in a certificate, local storage facilities or LDAP-accessible directories.

PKIF permits the use of the same public key for CRL signature verification as the one used for verifying the signature on the certificate, but does not mandate it. In other words, PKIF will develop and validate certification paths to CRL signers where necessary.

5.8 Symmetric key encryption and decryption

PKIF provides functionality to perform symmetric key encryption and decryption using algorithms including DES and Triple DES.

5.9 ASN.1 encoding/decoding

PKIF performs decoding of objects in support of processing related to X.509, RFC3280, OCSP and CMS. PKIF performs encoding of objects in support of processing related to OCSP and CMS.

5.10 Roles, User Data, and TSF Data

PKIF is a toolkit used by application developers to incorporate secure PKI functionality into an application; PKIF has only one role: user. The user is considered to be the application using PKIF, or, to provide a human definition, the application developer.

TOE user data is defined as any data that is passed to or returned from PKIF. This includes data that is encrypted, decrypted, signed, and verified or information used in support operations on such data. Trust anchors, certificates, CRLs, OCSP requests and responses are also user data.

Note that, for PKIF, the TOE environment performs the identification and authentication (I&A) functions. Therefore, data associated with I&A is not considered TSF data, since it is not within the TOE boundary. Similarly, private keys are managed by FIPS 140-2 validated cryptographic modules present in the environment and are not considered TSF data. Thus, there are no TSF data in PKIF.

5.11 TOE Environment Description

PKIF is intended for use with Microsoft Visual C++ .NET 2002. All references to IDE dialogs, property pages, fields, etc. assume use of Microsoft Visual C++ .NET 2002.

PKIF is designed to operate with any CAPI-compatible cryptographic module, including middleware that interacts with Common Access Cards (CAC). CACs are cryptographic modules that are validated at FIPS 140 series Level 1 or greater. Cryptographic modules may perform one or more of the following: key pair generation, digital signature generation and verification, encryption, decryption, secure hash, random number generation, Hash-based Message Authentication Code (HMAC) and/or other required cryptographic functions.

Certificates and revocation status information, i.e., CRLs or OCSP responses, are included in the environment and are available when requested by PKIF.

PKIF is intended for use on PCs running Windows 2000. Windows 2000 includes LDAP and HTTP client functionality. Windows 2000 includes a CAPI-compatible FIPS 140 Level 1 validated cryptographic module. In addition, the configuration includes the ActivCard CAC CSP.

The hardware configuration includes any PC with at least 128MB RAM, 20 GB hard drive, display, keyboard, mouse and, optionally, a smart card reader and CAC.

PKIF will build and validate certification paths to any trust anchor. For example, in order to use PKIF with a DoD-issued CAC, the DoD Class 3 Root needs to be included as one of the trust anchors in CAPI or otherwise made available to PKIF as a trust anchor. While operational DoD systems have the requirements to delete various trust anchors except for those required by Microsoft, the evaluated configuration does not depend on that requirement.

When using a CAC, the user certificates associated with the private keys stored on the CAC must be imported into a CAPI certificate store and associated with the CAC.

PKIF can be configured to search an application specified LDAP-accessible directory or to retrieve certificates and CRLs from HTTP or LDAP URLs included in certificates. To obtain information via HTTP or LDAP, the workstation must have network connectivity and access to the servers of interest. The evaluated configuration permits sufficient network connectivity.

The TOE environment provides the ActivCard CAC CSP module, which is installed by an administrator using the ActivCard installation application program.

PKIF is installed by an administrator of the workstation on which PKIF is being installed. PKIF is installed using the PKIF installation application.

The Windows 2000 OS environmental component provides Identification and Authentication (I&A) services. I&A is useful for access control of resources managed by Windows including files, folders, CAPI certificate stores, private keys, and audit logs (audit logs are maintained in a specific folder in the file system hierarchy). Windows 2000 I&A is used for identifying the event-causing subject and for identification of roles.

6 Documentation

The following is a list of the end-user documentation that was used to support this evaluation:

- *Public Key Infrastructure Framework (PKIF) Version 1.2 PKIF Security Target Version 1.63*, December 6, 2005 [ST].
- *U.S. Government Family of Protection Profiles for Public Key-Enabled Applications, Version: 2.61*, July 31, 2004 [PP].
- *PKIF Usage Guide*, Version 1.1.12.1, March 2005 [PKIFUG].
- *Public Key Infrastructure Framework (PKIF) Delivery, Installation, Generation and Start-up Procedures*, Version 1.2, August 2, 2004 [IGS].

7 IT Product Testing

7.1 Developer Testing

The vendor testing covered all of the security functions described in section 6 of the ST. The evaluators verified that each SFR had a corresponding test case and verified that the vendor testing approach was adequate to test and verify the behavior of the SFRs. A determination that the testing was systematic is supported by the evaluators demonstrating complete coverage for expected SFR behaviour. The correspondence between the test coverage and the functional specification was verified. Test coverage was verified for all TSF interfaces.

The evaluation team executed independent tests to verify proper behavior of the SFRs by first executing all vendor test cases. The TOE was installed using the vendor-supplied documentation. The Vendor test cases were performed and verified. These tests were executed using the developer test plan step-by-step guidance. The output from each of the tests was recorded by the test team as evidence. The test report demonstrates complete coverage for all TSF interfaces by the developer tests.

The evaluation team determined that the developer's actual test results matched the vendor's expected results.

7.2 Evaluation Team Independent Testing

The evaluation team ensured that the TOE performed as described in the design documentation and demonstrated that the TOE enforces the TOE security functional requirements. Specifically, the evaluation team ensured that the developer test documentation sufficiently addresses the security functions as described in the functional specification and high level design. The evaluation team performed the developer's test suite and devised an independent set of team tests and penetration tests. The team testing included supplemental tests using the vendor's test harness as well as modifications to the test harnesses to exercise additional aspects of the TSF.

Independent testing used the design documentation and user documentation to devise additional test opportunities. Team testing was performed to ensure that inconsistent policy settings, missing mediators, uninitialized mediators, missing colleagues, and duplicate mediator initialization did not permit means of bypassing the TSF.

The TSF did not produce incorrect results for any of the tests (e.g. report a valid certification path when there were no valid paths). The TSF responses were consistent with the guidance. However, the evaluators had difficulty matching some of the responses to errors with the root causes of the errors. For example, the fault tolerance capabilities of the TSF prevent ValidatePath from producing an error when a cryptographic mediator is missing. Instead, ValidatePath reports those aspects of path validation that were successful (e.g. basic certificate checks). PKIF's fault tolerant behavior is both secure and described in the PKIF Usage Guide. However, it might complicate debugging applications. As a matter of convenience, additional warnings were added to the user guidance to facilitate debugging.

Thus, the evaluators found the supplied guidance adequate for the secure use of the TSF.

7.3 Strength of Function

The Strength of Function requirements were not applicable for this TOE. The threat level for the TOE authentication function is assumed to be **SOF-basic**. Strength of function applies only to non-cryptographic, probabilistic or permutational mechanisms. The SOF requirement applies to the identification and authentication functionality within the TOE and for this TOE the environment handles the identification and authentication functionality.

7.4 Vulnerability Analysis

The vendor searched for publicly known vulnerabilities specifically related to the TOE. No publicly-known vulnerabilities specific to the evaluated version of PKIF were found. The following public domain sources were used to identify and search for relevant vulnerabilities:

- Common Vulnerabilities and Exposures (CVE) (http://www.cve.mitre.org/)
- National Vulnerability Database (NVD) (http://nvd.nist.gov/)
- US-CERT Vulnerability Notes Database (http://www.kb.cert.org/vuls/)

The vendor's search for vulnerabilities also included search of the design documentation, user documentation, and source code. The vendor used automated tools to detect memory leaks and buffer overflow problems. Specifically:

- Rational PurifyPlus from IBM was used to detect coding errors, memory leaks, timing analysis and test coverage analysis. Further information on the tool can be found at http://www-306.ibm.com/software/awdtools/purifyplus

- RATS from Secure Software was used to perform static source code analysis to look for calls to functions whose incorrect use can create a vulnerability (for example buffer overflow). Further information on the tool can be found at http://www.securesoftware.com/resources/download_rats.html

The vendor performed extensive testing of the TOE against ASN.1 vulnerabilities discovered in 2003-4. The vendor used the United Kingdom National Infrastructure Security Coordination Center (NISCC) S/MIME test suite (consisting of over one million test cases) to ensure that PKIF is not vulnerable to ASN.1 attacks.

Assertions made that PKIF was not subject to hypothesized vulnerabilities was verified by confirming that the proposed flaw was addressed by the vendor's functional test suite. Specific assertions from the vendor's vulnerability analysis were selected for additional testing by the evaluator to confirm the quality of the vendor's vulnerability analysis. The evaluators used brainstorming to devise additional testing to verify proper TSF function. The evaluators' vulnerability analysis and penetration testing found no exploitable vulnerabilities and no residual vulnerabilities in the TOE in its intended environment.

The asset under attack is the information transiting the TOE. In general, the threat agent includes, but is not limited to: 1) people with TOE access who are expected to possess "average" expertise, few resources, and moderate motivation, or 2) failure of the TOE. The specific threats that the TOE is designed to counter are listed in section 3.2 of the ST.

8 Evaluated Configuration

The evaluated configuration includes the following:

- Visual C++ .Net
- Common Access Card
- PKIF software library
- ActivCard CAC CSP
- Microsoft CAPI

These components are hosted on Microsoft Windows 2000 Workstation.

The TOE does not include:

- underlying operating system (OS) software or hardware
- CAC card
- CAC CSP
- Microsoft CAPI
- Visual C++ .Net

Certificates, CRLs and OCSP responses are considered to be included in the environment and are available as part of the DoD PKI interface

9 Results of the Evaluation

The evaluation team performed the applicable Common Evaluation Methodology activities according to a CygnaCom proprietary methodology. As issues were raised during the evaluation process, observations were documented and provided to the sponsor for correction. Incremental ETRs were released to document the progress of the ST and TOE evaluations. The evaluation team provided rationale for each verdict as part of their final ETR, describing the steps that were executed for each work unit, including the source of information used to make an evaluation conclusion. The ETR provided detailed rationale for each evaluation decision.

A verdict for an assurance component is determined by the resulting verdicts assigned to the corresponding evaluator action elements. The evaluation was conducted based upon CC, Version 2.2; CEM, Version 2.2, and all applicable International Interpretations in effect on October 8, 2004.

The Evaluation Team assigned a Pass, Fail, or Inconclusive verdict to each work unit of each EAL 4 assurance component. For Fail or Inconclusive work unit verdicts, the Evaluation Team advised the developer of issues requiring resolution or clarification within the evaluation evidence. In this way, the Evaluation Team assigned an overall Pass verdict to the assurance component only when all of the work units for that component had been assigned a Pass verdict.

The evaluation determined that the product meets the assurance requirements of EAL 4. The details of the evaluation are recorded in the Evaluation Technical Report (ETR), which is controlled by CygnaCom. The security assurance requirements are displayed in the following table.

TOE Security Assurance Requirements

Assurance Component ID	Assurance Component Title
ACM_AUT.1	Partial CM automation
ACM_CAP.4	Generation support and acceptance procedures
ACM_SCP.2	Problem tracking CM coverage
ADO_DEL.2	Detection of modification
ADO_IGS.1	Installation, generation, and start-up procedures
ADV_FSP.2	Fully defined external interfaces
ADV_HLD.2	Security enforcing high-level design
ADV_IMP.1	Subset of the Implementation of the TSF
ADV_LLD.1	Descriptive low-level design
ADV_RCR.1	Informal correspondence demonstration
ADV_SPM.1	Informal TOE security policy model
AGD_ADM.1	Administrator guidance
AGD_USR.1	User guidance
ALC_DVS.1	Identification of security measures
ALC_FLR.1	Basic flaw remediation
ALC_LCD.1	Developer defined life-cycle model
ALC_TAT.1	Well-defined development tools
ATE_COV.2	Analysis of coverage
ATE_DPT.1	Testing: high-level design
ATE_FUN.1	Functional testing
ATE_IND.2	Independent testing – sample
AVA_MSU.2	Validation of analysis
AVA_SOF.1	Strength of TOE security function evaluation
AVA_VLA.2	Independent vulnerability analysis

The Validation Team agreed with the conclusion of the CygnaCom Evaluation Team, and recommended to CCEVS Management that an EAL4 augmented with ALC_FLR.1 certificate rating be issued for PKIF Version 1.2.

10 Validator Comments/Recommendations

The Validation team used vendor-supplied documentation to familiarize themselves with the TOE usage and environment. The Validator used a combination of communications with the evaluation team (largely via electronic mail), records review, and review of the final ETR results to verify the results of the evaluation team’s analysis. The evaluation team responded to Validator queries in a timely manner. No deficiencies were found in the execution of the CEM work units.

No significant issues were found during the validation. The evaluation team responded quickly to all validation team requests and observations.

The TOE is dependent upon the environment to perform base cryptographic services, which are provided by the Windows Operating System as CAPI modules. If the underlying cryptographic modules are compromised, the results of PKIF operations are undefined. While this threat is countered by environmental assumptions such as *AE.Configuration*, the user should be cautioned that care is necessary to ensure that the underlying operating system environment is not compromised, leading to a malfunction of the PKIF.

The user should also note that the functional requirements FPT_RVM.1 and FPT_SEP.1 are levied on the IT environment (operating system). These requirements provide protection of the PKIF-based application as a whole and the PKIF DLL.

11 Security Target

The security target for PKIF is contained within the document *Public Key Infrastructure Framework (PKIF) Version 1.2 PKIF Security Target Version 1.63* dated December 6, 2005 [ST]. The ST is compliant with the *Specification of Security Targets* requirements found within Annex A of Part 1 of the CC [CCV2.2].

The document identifies the security functional requirements necessary to implement Access Control security policies. Additionally, the Security Target specifies the security assurance requirements necessary for EAL 4.

12 Glossary

Acronym	**Expansion**
CA	Certification Authority
CAC	Common Access Card
CAPI	Microsoft Cryptographic Application Programming Interface
CC	*Common Criteria for Information Technology Security Evaluation.* [Note: Within this Validation Report, CC always means Version 2.2, dated January, 2004.]
CCEVS	Common Criteria Evaluation and Validation Scheme
CCIMB	Common Criteria Interpretations Management Board
CCTL	Common Criteria Testing Laboratory
CEM	Common Evaluation Methodology
CMS	Cryptographic Message Syntax protocol
CPV	Certification Path Validation
CRL	Certificate Revocation List
CRLDP	CRL Distribution Point
DES	Data Encryption Standard
DN	Distinguished Name

DLL	Dynamic Link Library
DoD	Department of Defense
EAL	Evaluation Assurance Level
ETR	Evaluation Technical Report
FIPS	Federal Information Processing Standard
GMT	Greenwich Mean Time
HMAC	Hash-based Message Authentication Code
I&A	Identification and Authentication
IETF	Internet Engineering Task Force
IT	Information Technology
LDAP	Lightweight Directory Access Protocol
NIAP	National Information Assurance Partnership
OCSP	On-line Certification Status Protocol
OS	Operating System
PKE	Public Key Enabled
PKEPP	Public Key Enabled Protection Profile
PKI	Public Key Infrastructure
PKIF	Public Key Infrastructure Framework
PKIX	Public Key Infrastructure Working Group, IETF
PP	Protection Profile
RFC	Request for Comments
RSA	Rivest, Shamir, and Adelman
SF	Security Function
SFP	Security Function Policy
SOF	Strength of Function
ST	Security Target
TA	Trust Anchor
TOE	Target of Evaluation
TSC	TSF Scope of Control
TSF	TOE Security Functions
TSFI	TSF Interface
USMC	United States Marine Corps

13 Bibliography

URLs

- Common Criteria Evaluation and Validation Scheme (CCEVS): (http://www.niap.nist.gov/cc-scheme).

CCEVS Documents

[CCV2.2] *Common Criteria for Information Technology Security Evaluation*, CCIMB-2004-01-002, Version 2.2, January 2004.

[CEMV2.2] *Common Methodology for Information Technology Security Evaluation*, Version 2.2, Part 2: Evaluation Methodology, January 2004.

[CCEVS3] *Guidance to Validators of IT Security Evaluations*, Version 1.0, February 2000.

[CCEVS4] *Guidance to Common Criteria Testing Laboratories*, Draft, Version 1.0, March 2000.

Other Documents

[IGS] Public Key Infrastructure Framework (PKIF) Delivery, Installation, Generation and Start-up Procedures, Version 1.2, August 2, 2004.

[PKIFUG] *PKIF Usage Guide*, Version 1.1.12.1, March 2005.

[PP] *U.S. Government Family of Protection Profiles for Public Key-Enabled Applications, Version: 2.61*, July 31, 2004.

[RFC_3369] Cryptographic Message Syntax (CMS), RFC 3369, August 2002.

[ST] *Public Key Infrastructure Framework (PKIF) Version 1.2 PKIF Security Target Version 1.63,* December 6, 2005.