Security Target Document

Passport Certificate Server Ver. 4.1.1

Prepared for:

Common Criteria EAL2 (augmented)

30 April 2002

2225 Sheppard Ave, Suite 1700

Toronto, Ontario, Canada

M2J 5C2

TEL: 416-756-2324

FAX: 416-756-7346

Info@dvnet.com

www.dvnet.com

TABLE OF CONTENTS

1 Introduction

1.1 Security Target Identification

Title: Diversinet Passport Certificate Server, version 4.1.1

Registration: <to be filled in by registry>

Keywords: Public Key Infrastructure, Authentication.

1.2 Security Target Overview

This Security Target (ST) document describes the Diversinet Passport Certificate Server 4.1.1 as a Target of Evaluation (TOE) under the Common Criteria (CC), Version 2.1, for a claimed Evaluation Assurance Level (EAL) 2 (augmented).

The Diversinet Passport Certificate Server, version 4.1.1, is a Public Key Infrastructure (PKI) product that creates, manages and distributes public key certificates. The Passport Certificate Server is a software product that can be installed and operated on a variety of hardware platforms subject to the developer's specifications. This product is designed to provide transparent public key certificate services to users and their applications

This Security Target was written by Alfred Arsenault of Diversinet, and has a publication date of 30 April 2002.

1.3 Common Criteria Conformance

The TOE conforms to Part 2, Security Functional Requirements, and Part 3, Security Assurance Requirements, of Version 2.1 of the Common Criteria.

2 TOE Description

2.1 Product Deployment

The Diversinet Passport Certificate Server version 4.1.1 provides the infrastructure components for a secure wireless Public Key Infrastructure, or PKI. Its typical use is shown in Figure 2.1, "Typical Deployment of Passport Certificate Server.

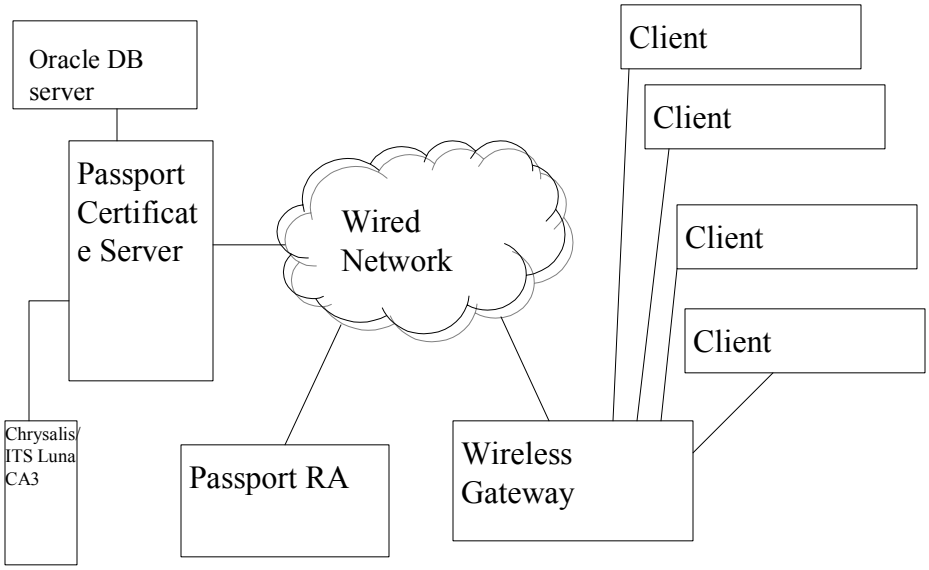

Figure 2-1: Typical Deployment of Passport Certificate Server

The TOE being described in this Security Target consists of the Passport Certificate Server.

As shown in Figure 2-1, the Passport Certificate Server (CS) is a computer typically connected to a wired network, such as the Internet or a corporate Intranet or Extranet. The CS is supported by a database (typically, Oracle 8i), on the same or another computer; and a hardware cryptographic module (typically, the Chrysalis/ITS Luna CA3), connected to the computer via a board.

The Passport RA is also typically connected to the same wired network as the CS.

Clients are mobile devices such as cell phones, two-way pagers, or personal digital assistants (PDAs), that transmit and receive messages through wireless means. Clients have software that understands the Diversinet-specified certificate formats and protocol messages, allowing them to communicate with the CS and RA.

In order for the clients to communicate with the CS and RA, they must first connect to a wireless gateway (sometimes referred to as a "base station"), which is responsible for converting the wireless signals into messages that are sent across the wired network.

2.2 Product Functions

The Diversinet Passport Certificate Server 4.1.1 provides system developers with a secure PKI capable of creation, management and retirement of public key certificates. The format of these certificates is

proprietary. The small size of the Diversinet Passport certificates is considered to be a significant advantage in embedded, high performance or constrained bandwidth systems over the X.509 version 3 certificate format. Functional Attributes in the certificate format can be employed by the user to store application-specific information. No identification and authentication information is contained in the Passport certificates. The main security functionality of the TOE is to provide secure management of certificates. This necessarily involves the use of cryptographic hardware and/or software to perform various security-related operations. The actual Passport certificate format is flexible enough to support a number of user-selectable algorithms. However, in this evaluation – and thus in this document – all public-key cryptography will be done using the RSA algorithm.

The Diversinet Passport Certificate Server 4.1.1 provides its clients with a secure information transfer service through its Secure Packet Exchange (SPEx) protocol. This protocol provides the assurance that no unauthorized access or integrity loss of certificates and other sensitive information occurs between the server and its client and associated administrator nodes.

In order to more fully understand the functions of the Passport CS, we will walk through a typical client registration. We will assume that the CS and RA have already been installed and are functioning normally. The steps are:

1. A user who wants a certificate contacts an RA operator and makes a request. This can be done through physical means; e.g., a person walks into a retail outlet and makes a request.

2. Once the RA has approved the request, the RA will generate a One-Time Secret (OTS) that will be used to authenticate the user later on. The RA will create and send a message to the CS, containing the name of the user and the OTS.

3. The CS will receive the message, "reserve" an ID in the system for this user, and send a message back to the RA containing this ID. The RA will provide this ID and the OTS to the user.

4. From his/her client device, the user will create and send a message to the CS, requesting that the certificate for which he/she is registered be created. This message will be authenticated using a Hashed Message Authentication Code (HMAC), using the OTS generated in Step 2 to generate the necessary key.

5. The CS will receive the message, generate and sign the certificate, and send the certificate back to the user.

(There are numerous variations to this process, but this is fairly typical and shows the important steps.)

Once a certificate has been issued, the CS will manage it during its lifecycle, by retiring it or deleting it when needed, or allowing it to expire at the agreed-upon time.

The other major function of the CS is to respond to a client request by validating the current status of a certificate and returning if appropriate a requested certificate.

2.3 Product Description

Now that we have provided context for the CS, we will describe in some level of detail its implementation.

2.3.1 Platform

As evaluated, the CS operates on an Intel Pentium-based computer running Microsoft Windows NT 4, Service Pack 5 or higher. The CS uses the Oracle 8i database for storage and retrieval of security-relevant

and other information. For cryptographic needs, the CS uses RSA's BSAFE 4.3.1. This is graphically illustrated in Figure 2-2, "High-level Overview of Passport Certificate Server Architecture."

2.3.2 Roles

The Passport system supports three different roles: security officer, administrator, and user. The security officer role is created when the Certificate Server is installed, and there is only one security officer in the system at a time. The security officer operates directly from the console of the computer hosting the CS. The security officer role creates one or more administrators. Administrators can operate RA's, and manage user accounts. "Users", like administrators, are somewhat unusual in that they do not directly access the TOE; rather, they are the subjects of the certificates that the TOE issues and manages. That is, "users" typically will be using clients and sending and receiving messages from the Passport Certificate Server.

The following table summarizes the major functions of each of the three roles.

Role	Major Functions	Comments
Security Officer	set up Passport Certificate Server maintain Passport Certificate server create/delete Administrator accounts	Always connected to the Passport Certificate Server
Administrator	Reserve ID's for clients Assign aliases to ID's Assign Functional Attributes (FA's) to ID's Retire certificates Lock or delete ID's Display a list of ID's Display a list of retired certificates for an ID	
User	Request creation of (publish) certificate Retire certificate Lock or unlock ID	

2.3.3 Product Structure

In addition to the user-visible roles described above, the Passport Certificate Server is internally structured into different functions for security and performance purposes. Unlike many PKI products, the Passport

Certificate Server is responsible for both certificate signing and communicating certificate validity information to users. These functions have very different security implications. For example, compromise of a key used to sign certificates will allow an attacker to create any new certificate with any privileges or identity desired within the realm. On the other hand, compromise of a key used only to communicate with users about the status or validity of a certificate is not nearly as harmful. It might allow someone with a retired certificate to pretend that his certificate is still valid, but the degree of exposure is substantially less than loss of a key used to sign certificates.

For this reason, the Passport Certificate Server v. 4.1.1 is internally structured into two major functions, the Certificate Authority (CA) and the Certificate Server (CS). These functions are logically separate, and each has its own public/private key pair. Because of the security concerns, and because certificate signing does not cause major performance concerns, the CA's private key is stored on the Chrysalis/ITS LunaCA3 token, and is only unlocked by entering the PIN directly on that device.

The CS key is used to sign and/or decrypt communications with users, either certificate registration requests ("publish" requests) or certificate validation queries ("fetch" requests). Because publish and fetch communication with clients is expected to happen frequently, there are severe performance implications of putting the CS private key on the LunaCA3. Since the security implications of a compromise are also lower than for the CA key, the CS private key is kept stored in encrypted form on the Certificate Server device, and is made available when the Administrator or security officer passphrase is entered correctly.

Note that the separation of the CA and CS functions is not visible to users; it is simply an internal structuring done to improve security and performance.

2.4 Scope of Evaluation

Applications that use the Diversinet Passport Certificate Server are not considered to be within the scope of the TOE under evaluation. Likewise, the Oracle 8i database that acts as a repository for certificates created by the CS is outside the scope of this evaluation. At the time this ST was prepared, Oracle7 server 7.3.4.0 has obtained an evaluation of EAL3 under the CC, but this evaluation has not been appealed to or utilized in any way in this ST document. Nor are any results applicable to an evaluation of Microsoft Windows NT or any related product.

As noted, the CS uses cryptographic functions – both symmetric and public-key – to provide security services that meet the requirements specified in this ST. The CS uses the RSA BSAFE Crypto-C toolkit to provide those cryptographic functions. Note that RSA BSAFE Crypto-C Toolkit version 4.31 has been awarded certificate #89 under the FIPS 140-1 Cryptographic Module Validation Program. This module meets FIPS 140-1 level 1 when running on Windows NT.

As noted, the CS uses cryptographic functions – both symmetric and public-key – to provide security services that meet the requirements specified in this ST. However, it does not provide a generic encryption service for user data. That is, users are not expected to call upon Passport Certificate Server functions to encrypt arbitrary data, e.g., encrypt bulk data for storage on disks.

Figure 2-3, "TOE Scope of Control", demonstrates graphically what is within the TOE scope and what is outside the TOE Scope.

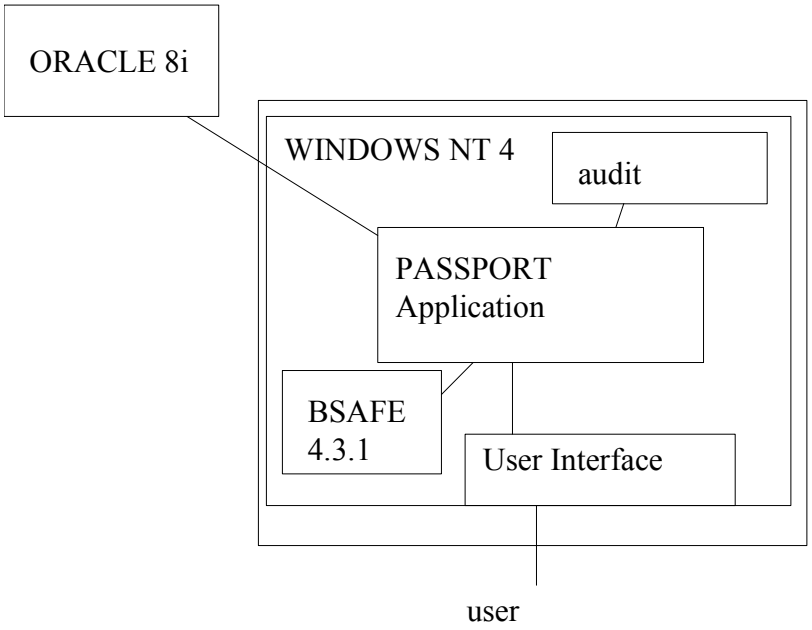

Figure 2-2: High-level overview of Passport Certificate Server Architecture

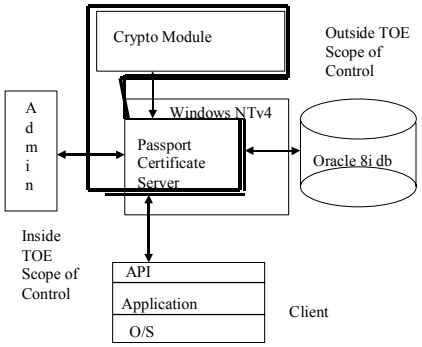

Figure 2-3: TOE Scope of Control

2.5 Diversinet Security Policies

Throughout this document, Security Functional Requirements will refer to "the Diversinet Access Control Policy" and "the SPEx Information Flow Policy". These are the policies enforced by the TOE as designed and implemented. These policies are as follows:

2.5.1 Diversinet Access Control Policy

The Diversinet Passport Certificate Server v.4.1.1 complies with the following Access Control Policy:

Roles are:

1. User
2. Administrator
3. Security Officer

Subjects are:

1. third party entity
2. user(s) (instantiations of User role, member of set: Users)
3. administrator(s) (instantiations of Administrator Role, member of set: Administrators)
4. security officer (unique instantiation of Security Officer Role)
5. server (as a signing entity)
6. root (as a signing entity)

Objects are:

1. User Key Pair comprising
    a. Public Key of user
    b. Private Key of user
2. User Certificates, containing the security attributes of users, comprising:
    a. Version information
    b. Key Information describing the nature of the public key, specifying cryptographic provider, key size and other provider-specific options
    c. Realm
    d. ID
    e. Sequence
    f. Functional Attribute (application-definable value)
    g. Public Key of user
    h. Timestamp
    i. Signature Options

j. Digital Signature (generated with the server's private key)

Login access consists of entry of a passphrase, known only to, and in one-to-one correspondence with, an Administrator.

Private Keys are accessible only to a single user, and are in one-to-one correspondence with the set Users. Private keys are encrypted and stored on the client node accessible to their associated user

Certificates have unlimited access.

Users may lock their own Ids

Administrators may lock, unlock, assign alias, and assign functional attribute for any user ID.

2.5.2 SPEx Information Flow Policy

The Secure Packet Exchange (SPEx) protocol complies with the following policy requirements:

1. All SPEx transmissions are authenticated between client and server.

2. All SPEx transmissions are confidential and protected against electronic eavesdropping.

3. SPEx IP packets are time-stamped and digitally signed to protect against unauthorized modification, substitution, insertion of unauthorized packets, and replay of packets.

4. Transmissions between client and server on the following list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects are protected against unauthorized disclosure:

    a. A new client subject may transmit a client-generated one-time secret (OTS) to the server subject encrypted with the server's public key (in open mode) or to an Administrator (acting as Registration Authority in closed mode);

    b. The server subject may generate a user ID and transmit this object to the new client subject (in open mode) or to an Administrator (Registration Authority in closed mode).

3 TOE Security Environment

The TOE security Environment provides a clear and consistent definition of the "security problem" that the TOE and its environment address.

3.1 Assumptions

This section identifies the assumptions that are made about the environment in which a Passport Certificate Server will operate.

3.1.1 Physical

A.NoObservation - The physical environment allows users to enter passwords without being directly observable by other users or potential threat agents

A.SecureArea – The certificate server system will be kept in a physically secure area. "Physically secure" means that only authorized personnel – security officers – can get unsupervised access to the machine at any time.

3.1.2 Personnel

A.NoEvilAdmin - The selection of personnel for administrative roles with respect to the TOE's deployment and use in the organization must include a proper background check of the individual or be justified by mitigating circumstances that provide the organization with the assurance that administrators will demonstrate competence in their duties and not deliberately misuse or subvert the TOE for non-secure, fraudulent or other improper purposes.

A.NoSharing – Users will obey organizational rules and not share passwords.

3.1.3 Connectivity

A.Oracle – The Oracle 8i database that acts as a repository for certificates created by the Diversinet Passport Certificate Server will be available with full database administration support. The Oracle database will always be available and configured correctly.

3.2 Threats

The list of threats that target the assets and services that the TOE is protecting is as follows:

T.Masquerade - A hostile agent or user of an application/service that depends on the TOE for authentication of origin may create an unauthorized certificate or modify the information content of an existing certificate for the purpose of masquerading as an authorized user.

T.Repudiation - A hostile user of an application/service that depends on the TOE for authentication of origin may attempt to repudiate the origin and time of a transaction involving a distributed application.

T.KeyTheft - A hostile agent may acquire a legitimate user's private key and use it to masquerade as that user.

T.Replay - A hostile agent may acquire a transmitted certificate or other sensitive information and later replay it, in the hopes of defeating the authentication and allowed to perform some operation on the system.

T.Import - A hostile agent may attempt to import unauthorized certificates into the server certificate database

T.Roles - A hostile user may attempt to create a privileged role (Security Officer or Administrator) over which he/she would control the PKI for the perpetration of other security attacks.

T.Eavesdrop - A hostile agent may employ sniffers or other monitoring devices to capture information flows between client and server, or between the server and the certificate database, with the object of acquiring critical security information such as the private keys of the Security Officer, Administrators or users.

The complete definition of each of these threats, in terms of its agents, attacks, and assets, is given in the following table:

Threat	Agent	Attack Method	Asset Attacked
T.Masquerade	authorized user of a client application, or a third-party gaining some level of physical access to some part of the system. moderately skilled network user has commercially-available and shareware tools to use	requesting a certificate with values he is not authorized to have; modifying an existing valid certificate to change the name or increase the privileges.	Certificates issued by the System
T.Repudiation	authorized user of a client application, or a third-party gaining some level of physical access to some part of the system. moderately skilled network user has commercially-available and shareware tools to use	Repudiating a transaction; i.e., denying that he participated in a transaction in which he did in fact participate.	Applications using the PKI; and thus indirectly the credibility of the PKI.

Threat	Agent	Attack Method	Asset Attacked
T.KeyTheft	authorized user of a client application, or a third-party gaining some level of physical access to some part of the system. moderately skilled network user has commercially-available and shareware tools to use	Learning the private key of a legitimate system user, allowing the attacker to successfully impersonate that user.	Private key of a user
T.Replay	authorized user of a client application, or a third-party gaining some level of physical access to some part of the system. moderately skilled network user has commercially-available and shareware tools to use	Capturing and replaying valid information (e.g., a certificate or a transaction) in an attempt to defeat system identification/ authentication and accountability policies	Ability to authenticate to system
T.Import	authorized user of a client application, or a third-party gaining some level of physical access to some part of the system. moderately skilled network user has commercially-available and shareware tools to use	Importing invalid certificate information into the database, so that later certificates and requests will appear to be valid.	Integrity of the certificate database.

Threat	Agent	Attack Method	Asset Attacked
T.Roles	authorized user of a client application, or a third-party gaining some level of physical access to some part of the system. moderately skilled network user has commercially-available and shareware tools to use	Attempting to gain Security Officer or Administrator privileges on the system, to facilitate carrying out other attacks later on.	Privileges to perform security-critical operations.
T.Eavesdrop	authorized user of a client application, or a third-party gaining some level of physical access to some part of the system. moderately skilled network user has commercially-available and shareware tools to use	Using a sniffer or other monitoring tool to capture sensitive information being sent across the network.	Sensitive information being passed across the network

4 Security Objectives

4.1 Security Objectives for the TOE

The Security Objectives of the TOE comprise the following:

O.AccessControl - The TOE must provide its users with the means of controlling and limiting access to the objects and resources they own or are responsible for, on the basis of individual users or identified groups of users, and in accordance with the set of rules defined by the TOE access control policy.

O.Accountability – The TOE must enforce accountability for the actions of all users. Security-relevant actions must be audited, and associated with the identity of the user who caused the action to occur.

O.Import – The TOE must provide a protection against unauthorized importation of certificates to the server certificate database. The digital signatures of the server must distinguish authentic certificates from invalid certificates.

O.NonRepudiation – The TOE must provide a mechanism to authenticate third party application transactions and services through secure public key distribution.

O.Replay – The TOE must eliminate the potential for replay attacks over internal transmissions of sensitive information by the use of a protocol in which digitally signed time stamps and integrity checking is enforced.

O.CERT_VALS.CERT_CHANGE The TOE must provide control over the user functions and values contained within the certificate structure.

O.CERT_VALS.DEFAULT_OVERRIDE – The TOE must only allow those with the Administrator role to override configured default values for certificates.

O.CERT_VALS.RESTRICT_CHANGE – The TOE must ensure that only those with the Administrator role can change values in certificates.

O.CERT_VALS.SEC_VALS – The TOE must only allow secure values for user data in certificates.

O.KeyManagement – The TOE must provide for the management of cryptographic keys in terms of generation and destruction.

O.Key_Distr – The TOE must provide for the secure distribution of cryptographic keys.

O.KeyAccess –The TOE must restrict access to private keys to those who are authorized to use them.

O.Roles.ID – The TOE must provide and maintain the roles: Security Officer, Administrator and User. The TOE must be able to associate users with those roles.

O.Roles.Create - All roles require hierarchical approval and control for the creation of roles of lower privilege.

O.OneTimeSecret - The TOE must provide that the use of one-time secrets must be enforced for generating a Message Authentication Code (MAC) when a user obtains an ID and initial certificate. *(The TOE must provide that the quality metrics of size and uniqueness of one-time secrets be enforced.)*

O.TimeStamp - The TOE must provide that the TSF use reliable internal time stamps to support certificate management functions.

O.CertConfidentiality - The TOE must provide that the TSF protect transmissions of user certificates between the server and client from unauthorized disclosure to threat agents.

O.Cert_Integ - The TOE must provide that the TSF protect transmissions of user certificates from modification, deletion, insertion and replay errors, to detect such errors and allow recovery.

O.Trans_conf - The TOE must provide that the TSF be able to protect sensitive information from unauthorized disclosure while it is being transmitted across a network.

O.TRANS_INTEG.SPEX – The TOE must provide that the TSF be able to detect modification of user certificates during transmission.

O.UserID - The TOE must provide that identification (by selection of user ID from the login form) must be done prior to any other TSF actions, such as encryption / decryption of protected data.

O.UserAuth - The TOE must provide that the user cannot perform actions such as encryption / decryption of protected data (other than selection of user ID from the login form) prior to authentication of the user's identity.

O.AuthFeedback - The TOE must provide that the authentication feedback to the user be limited, allowing specifically that the characters of the user authentication string could be represented by asterisks (*).

4.2 Security Objectives for the Environment

The Environmental Security Objectives comprise the following:

O.CryptoKeyProtection - Procedural and physical measures should be taken to prevent unauthorized individuals from gaining access to the TOE user private keys and keys used for encryption of user private keys.**O.NO_EVIL** - The system administration roles must be staffed by adequately trained, responsible and honest individuals who are not motivated to disable, degrade or subvert the operation of the TOE in the environment for personal gain or other purposes that contradict the security policies of the organization

O.NO_OBSERV - The physical environment allows users to enter passwords without being directly observable by other users or potential threat agents. The environment does not have concealed or visible video capture devices such as closed circuit TV equipment or video camera equipment that could be used to capture a user's key strokes at a distance. The use of video recording equipment within line of sight of the PC or workstation hosting protected drives/partitions must be sanctioned by the security officer and all video information resulting from its operation protected from unauthorized access.

O.PHYS_ACC - The PC or workstation hosting sensitive private key material must be located in a lockable cabinet or room, or be logged-out or powered down when unattended. No unauthorized person – i.e., no one except those authorized to fill the security officer role – may ever have unsupervised access to the computer on which the CS executes.

O.PWD_SHARE - The sharing of passwords, that provide read or modify access to private keys, among users should be forbidden or, if required to enforce role or group access, strictly confined to the users who hold the specified organizational role or are members of the specified group authorized to access the information assets protected by the shared password

O.SEC_AWARE - Users should be properly trained in Organizational security policy and have awareness of security procedures

O.UNATTEND - Administrators must be trained on correct procedures to follow when their PCs and workstations are unattended, and password-enabled screen savers and similar protective software should be used if their use is warranted

O.SEC_LINK - The communications link between the server and the database must be protected from unauthorized disclosure, modification or playback, or loss of bandwidth. This may be accomplished through adequate physical security and access control as well as sufficient maintenance of the system

5 IT Security Requirements

5.1 TOE Security Functional Requirements

This section contains the security functional requirements for the TOE. The following CC Part 2 Components are referenced, with definitions reproduced verbatim or completed where required. Completed definition text (i.e., added text not defined by the CC) is indicated below by *italics*.

5.1.1 Audit

FAU_GEN.1.1

The TSF shall be able to generate an audit record of the following auditable events;

a) Start-up and shutdown of the audit functions
b) All auditable events for the *minimum* level of audit; and
c) Publication of Certificates, Assignment of Alias, Deletion of all Retired Certificates, Get First ID Information, Get ID Information by Alias, Get Next ID Information, Get Previous ID Information, Lock ID, Set FA, Get Last ID Information, Delete Retired Certificate, Delete ID, Get First ID Information by FA, Get ID Information by ID, Get Next ID Information by FA, Get Previous Next ID Information by FA, Reserve ID, Unlock ID, Get Last ID Information by FA.

FAU_GEN.1.2

The TSF shall record within each audit record at least the following information:

a) Date and time of event, type of event, subject identity, and the outcome (success or failure) of the event; and

b) For each audit event type, based on the auditable event definitions of the functional components included in the ST: *Publication of Certificates, Assignment of Alias, Deletion of all Retired Certificates, Get First ID Information, Get ID Information by Alias, Get Next ID Information, Get Previous ID Information, Lock ID, Set FA, Get Last ID Information, Delete Retired Certificate, Delete ID, Get First ID Information by FA, Get ID Information by ID, Get Next ID Information by FA, Get Previous Next ID Information by FA, Reserve ID, Unlock ID, Get Last ID Information by FA*.FAU_SAR.1.1

The TSF shall provide *Security Officers* with the capability to *read message type, event type* from the audit records.

FAU_SAR.1.2

The TSF shall provide the audit records in a manner suitable for the user to interpret the information

5.1.2 Communications

FCO_NRO.1.1

The TSF shall be able to generate evidence of origin for transmitted *certificates* at the request of the *recipient*.

FCO_NRO.1.2

The TSF shall be able to relate the client ID (contains issuer's ID), realm ID, public key, signature algorithm(s) of the originator of the information, and the certificate serial ID, sequence identifier, client ID (contains issuer's ID), realm ID, functional attribute, public key, signature algorithm(s) of the information to which the evidence applies.

FCO_NRO.1.3

The TSF shall provide a capability to verify the evidence of origin of information to *recipients* given that *the information is digitally signed or protected with an HMAC.*

5.1.3 Cryptographic Support

Cryptographic Key Management

FCS_CKM.1.1 The TSF shall generate cryptographic keys in accordance with *RSA* that are *1024 bits* long that meet the following: *RSA PKCS#1 and FIPS 186-2*.

FCS_CKM.4.1 The TSF shall destroy cryptographic keys in accordance with a specified cryptographic key destruction method: *by deleting the encrypted representation of the key from the Windows NT registry, and deleting and overwriting any copies of the key in active memory* that meets the following**:** FIPS 140-1.

Cryptographic Operation

FCS_COP.1.1 The TSF shall *encrypt, decrypt, sign and verify signatures* using *1024-bit RSA* keys in accordance with *RSA PKCS#1 and FIPS 186-2*

5.1.4 User Data Protection

Access Control Policy

FDP_ACC.2.1

The TSF shall enforce the Diversinet Access Control Policy on *Subjects: Security Officer, Administrator and User, and objects: certificates* and all operations among subjects and objects covered by the *Diversinet Security Functional Policy*

FDP_ACC.2.2

The TSF shall ensure that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP.

FDP_ACF.1.1

The TSF shall enforce the Diversinet Access Control Policy to objects based on *type of object.*

FDP_ACF.1.2

The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: *a) the private key of the appropriate CA must be used to sign user certificates b) the public key assigned to the User must be contained in his/her certificate.*

FDP_ACF.1.3

The TSF shall explicitly authorize access of subjects to objects based on the following additional rules: *a) an Administrator can revoke a certificate b) a User can revoke his/her certificate.*

FDP_ACF.1.4

The TSF shall explicitly deny access of subjects to objects based on the rules: *a) The private key of a user cannot be accessed by any other user*.

Data Authentication

FDP_DAU.1.1 Basic Data Authentication

The TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of *public key certificates*.

FDP_DAU.1.2 Basic Data Authentication

The TSF shall provide *users* with the ability to verify evidence of the validity of the indicated information

Export to outside TSF control

FDP_ETC.2.1 Export of user data with security attributes

The TSF shall enforce the Diversinet Access Control Policy when exporting user data, controlled under the SFPs, outside of the TSC.

FDP_ETC.2.2 Export of user data with security attributes

The TSF shall export the user data with the user data's associated security attributes.

FDP_ETC.2.3 Export of user data with security attributes

The TSF shall ensure that the security attributes, when exported outside the TSC, are unambiguously associated with the exported user data.

FDP_ETC.2.4 Export of user data with security attributes

The TSF shall enforce the following rules when user data is exported from the TSC: Certificates and encrypted private keys may be exported.

Information flow control policy

FDP_IFC.1.1 Subset information flow control

The TSF shall enforce the SPEx Information Flow Policy on the following list of subjects, information, and operations that cause controlled information to flow to and from controlled subjects covered by the SPEx Information Flow Policy*:*

a) A new client subject (in open mode) or an Administrator subject (in closed mode) may transmit a client-generated one-time secret (OTS) to the server subject encrypted with the server's public key (in open mode)

b) The CS subject may generate a user ID and transmit this object to the new client subject (in open mode) or to an Administrator (Registration Authority in closed mode).

Information flow control functions

FDP_IFF.1 Simple Security Attributes

FDP_IFF.1.1 The TSF shall enforce the SPEX Information Flow Policy based on the following types of subject: *User, Administrator, and Security Officer;* and information security attributes: *whether a message is signed or protected by an HMAC.*

FDP_IFF.1.2 The TSF shall permit an information flow between a controlled subject and controlled information via a controlled operation if the following rules hold: *if the message is signed or HMACed in accordance with the SPEx Information Flow Policy.*

FDP_IFF.1.3 The TSF shall enforce *no other information flow control rules.*

FDP_IFF.1.4 The TSF shall provide *no other information flow capabilities.*

FDP_IFF.1.5 The TSF shall explicitly authorize an information flow based on the following rules: *a signed message from a User or Administrator, or from the Security Officer, will be permitted to pass. An HMAC'ed message from a User or Administrator will be accepted if it is part of a registration or update Root Key Set operation.*

FDP_IFF.1.6 The TSF shall explicitly deny an information flow based on the following rules: *any message that does not have a valid signature or HMAC will be rejected.*

Stored Data Integrity

FDP_SDI.1.1 Stored data integrity monitoring

The TSF shall monitor user data stored within the TSC for *unauthorized changes* on all objects, based on the following attributes: digital signature of certificate.

FDP_UCT.1.1

The TSF shall enforce the SPEx Information Flow Policy to be able to *transmit* objects in a manner protected from unauthorized disclosure.

5.1.5 Identification and Authentication

FIA_ATD.1.1

The TSF shall maintain the following list of security attributes belonging to individual users: *client ID (contains issuer's ID), realm ID, public key*

FIA_SOS.1.1

The TSF shall provide a mechanism to verify that secrets meet the *metric: chance of a random guess finding the correct secret is less than 1 in 1,000,000.. The Strength of Function rating for passphrases shall be SOF-Medium, and the Strength of Function rating for One-Time Secrets shall be SOF-Basic.*[1]

[1] Note: Strength of function analysis and calculations are described in detail in the separate document, "Analysis of the Strength of Functionality of Security Mechanisms in Diversinet Passport Certificate Server v.4.1.1." Readers should note that, as explained in that document, no assessments of cryptographic

FIA_UAU.2.1

The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

FIA_UAU.7.1

The TSF shall provide only *asterisks to be displayed* to the user while the authentication is in progress.

FIA_UID.2.1

The TSF shall require each user to identify itself before allowing any other TSF-mediated actions on behalf of that user.

5.1.6 Security Management

FMT_MOF.1.1

The TSF shall restrict the ability to *disable and enable* the functions *user reserve ID* and *creation of administrators* to the *Security Officer role*.

FMT_MSA.1.1

The TSF shall enforce the Diversinet Access Control Policy to restrict the ability to change_default and modify certificates to Security Officers.

FMT_MSA.2.1

The TSF shall ensure that only secure values are accepted for security attributes.

FMT_MSA.3.1

The TSF shall enforce the Diversinet Access Control Policy to provide *restrictive* default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2

The TSF shall allow the *Administrator* to specify alternative initial values to override the default values when an object or information is created.

FMT_MTD.1.1

The TSF shall restrict the ability to lock the user's ID to the user himself.

The TSF shall restrict the ability to lock the user's ID to the Administrator.

The TSF shall restrict the ability to unlock the user's ID to the Administrator.

FMT_MTD.3.1

---

algorithms are included in the Strength of Function claims made in this ST. That is, Strength of Function claims are independent of cryptography.

The TSF shall ensure that only secure values are accepted for TSF data, specifically One-Time Secrets and passphrases.

FMT_SMR.2.1

The TSF shall maintain the roles: *Security Officer, Administrator, User.*

FMT_SMR.2.2

The TSF shall be able to associate users with roles.

FMT_SMR.2.3

The TSF shall ensure that the conditions: *That the certificate database setting is accessible by the Security Officer; and that the Security Officer has knowledge of the database password* is satisfied.

5.1.7 Protection of the TSF

FPT_STM.1.1

The TSF shall be able to provide reliable timestamps for its own use.

5.1.8 Dependencies

Among the above IT Security Requirements are several that have dependencies on other requirements. The table below identifies the direct dependencies of each requirement. It also shows whether the requirement on which the dependency exists is included in the above list or excluded for the reasons explained.

Note that this table only shows direct dependencies. Many requirements also have indirect dependencies; e.g. FAU_SAR.1 depends on FAU_GEN.1, which depends on FPT_STM.1. Indirect dependencies are not shown directly in this table; they can be determined by following table entries. There is no requirement introduced by an indirect dependency that is not already either a direct system requirement or a requirement introduced by a direct dependency.

IT Security Requirement	Direct Dependencies	Remarks
FAU_GEN.1	FPT_STM.1	Included
FAU_SAR.1	FAU_GEN.1	Included
FCO_NRO.1	FIA_UID.1	Hierarchical component FIA_UID.2 is included
FCS_CKM.1	FCS_COP.1	Included
	FCS_CKM.4	Included
	FMT_MSA.2	Included
FCS_CKM.4	FCS_CKM.1	Included

IT Security Requirement	Direct Dependencies	Remarks
	FMT_MSA.2	Included
FCS_COP.1	FCS_CKM.1	Included
	FCS_CKM.4	Included
	FMT_MSA.2	Included
FDP_ACC.2.	FDC_ACF.1	Included
FDP_ACF.1	FDP_ACC.1	Hierarchical component FDP_ACC.2 is included
	FMT_MSA.3	Included
FDP_DAU.1	No Dependencies	
FDP_ETC.2.	FDP_IFC.1	Included
FDP_IFC.1	FDP_IFF.1	Included
FDP_IFF.1	FDP_IFC.1	Included
	FMT_MSA.3	Included
FDP_SDI.1.	No Dependencies	
FDP_UCT.1.	FTP_ITC.1	Included
	FDP_ACC.1	Hierarchical component FDP_ACC.2 is included
	FDP_IFC.1	Included
FIA_ATD.1	No Dependencies	
FIA_SOS.1.	No Dependencies	
FIA_UAU.2	FIA_UID.1	Included
FIA_UAU.7	FIA_UAU.1	Hierarchical component FIA_UAU.2 is included
FIA_UID.2.	No Dependencies	
FMT_MOF.1	FMT_SMR.1	Hierarchical component FMT_SMR.2 is included
FMT_MSA.1.	FDP_ACC.1	Hierarchical component FDP_ACC.2 is included
	FDP_IFC.1	Included
	FMT_SMR.1	Included

IT Security Requirement	Direct Dependencies	Remarks
FMT_MSA.2	FDP_ACC.1	Hierarchical component FDP_ACC.2 is included
	FDP_IFC.1	Included
	FMT_SMR.1	Included
FMT_MSA.3.	FMT_MSA.1	Included
	FMT_SMR.1	Hierarchical component FMT_SMR.2 is included
FMT_MTD.1	FMT_SMR.1	Hierarchical component FMT_SMR.2 is included
FMT_MTD.3	ADV_SPM.1	Included
	FMT_MTD.1	Included
FMT_SMR.2	FIA_UID.1	Included
FPT_STM.1	No Dependencies	

6 Security Assurance Requirements

6.1 Diversinet Security Assurance Measures

As part of its development and evaluation process, Diversinet employs a number of assurance measures. These assurance measures ensure that the quality and security provided by the Passport Certificate Server v.4.1.1 are known and are sufficient for our business.

The assurance measures employed by Diversinet that are relevant to the TOE are:

AM_Config: Diversinet employs a configuration management system that identifies each of the components of the TOE, and provides appropriate version numbers for each component. This process is documented in Diversinet Passport Certificate Server version 4.1.1 Configuration Management Plan.

AM_Delivery_install: Diversinet provides documentation for system users that describe in detail how the system is delivered when purchased, and how to install the system so that it is properly configured. In addition, automated installation, generation, and start-up procedures are provided by the Diversinet Passport Server version 4.1.1 installation Wizard. This is documented in Diversinet Certificate Server 4.0 User's Guide, 4th edition, and accompanying supplements.

AM_Design_documentation: Diversinet follows a design process that results in the development of a number of system design documents describing the product at various levels of abstraction. A functional specification describes the behavior of the product. A high-level design document describes the various modules and how they interact. A set of correspondence documents maps between the specification and the design to ensure that what is built is what was supposed to be built. An informal model of the product's security policy is developed to ensure that the access control and information flow policies to be enforced are properly understood and are internally consistent. The documents that describe this assurance measure include: Diversinet Passport Certificate Server v4.1.1High Level Design Document, Diversinet Passport Certificate Server v4.1.1Functional Requirements Specification, Diversinet Passport Certificate Server v4.1.1 Mapping from Descriptive High-level Design Document to Functional Specification, and Diversinet Passport Certificate Server v4.1.1 Correspondence Analysis – TOE Summary Specification to Functional Specification, and Diversinet Passport Certificate Server v4.1.1 Security Policy Model

AM_Guidance: Diversinet develops guidance on system operation addressed to the system users and administrators. This is contained in the Diversinet Passport Certificate Server v4.0 User's Guide and supplements. This is documented in Diversinet Certificate Server 4.0 User's Guide, 4th edition, and accompanying supplements.

AM_Testing: The Diversinet quality assurance program includes a strict testing component. The Diversinet Test Plan describes the testing procedures that are used to ensure that the product meets its functional requirements. In addition, independent testing will be done as part of the evaluation of this product. This is documented in Passport Certificate Server version 4.1.1 Security Functional Test Plan and Passport Certificate Server version 4.1.1 Evidence of Developer Test Coverage.

AM_Strength_of_function_evaluation: Security-critical aspects of the system, such as one-time secrets and passphrases, are analyzed to ensure that the chance of them being guessed by an attacker are acceptably low. This analysis is documented in Passport Certificate Server version 4.1.1 Strength of Function Analysis.

AM_Vulnerability_analysis: Security experts on the Diversinet staff analyze the design and implementation of the system to identify any potential vulnerabilities. Those vulnerabilities that are countered by the product design and implementation are identified. Other potential vulnerabilities not countered by technical means must be countered by the operational environment . Documentation is checked to ensure that customers are warned about the protections that their environments will have to

provide. This analysis is documented in Vulnerability Analysis of Diversinet Passport Certificate Server Version 4.1.1.

6.2 Security Assurance Requirements Rationale

The rationale for the Security Assurance Requirements against the Diversinet Security Assurance Measures is given in the table below. For each Diversinet Security Assurance Measure, a list of assigned Security Assurance Requirements is given, followed by an argument stating how each SAR maps to the security assurance measure in question.

Diversinet Security Assurance Measure	Security Assurance Requirement	Comments
AM_Config	ACM_CAP.2 Configuration Items	TOE releases are adequately identified with the version number.
AM_Delivery_install	ADO_DEL.1 Delivery procedures ADO_IGS.1 Installation, generation, and start-up procedures	The TOE Delivery procedures are described in the User guide installation notes. Automated installation procedures are adequate to ensure that the user starts the TOE within a secure configuration.
AM_Design_documentation	ADV_FSP.1 Informal functional specification ADV_HLD.1 Descriptive high-level design ADV_RCR.1 Informal correspondence demonstration ADV_SPM.1 Informal Security Policy Model	An informal functional specification is supplied for the TOE. A Descriptive high-level design for the TOE is available.A representational correspondence is supplied to connect the TOE summary specification to the informal functional specification of TSFs provided. An informal model of the Diversinet Security Policy enforced by the TSF is supplied.
AM_Guidance	AGD_ADM.1 Administrator guidance AGD_USR.1 User guidance	The administrator's guide is adequate to provide administrators with the required knowledge to securely configure and maintain the TOE within the environment. The User guidance is adequate to provide the user with the required knowledge to correctly perform login procedures and to provide security awareness of the TOE and its policies.

Diversinet Security Assurance Measure	Security Assurance Requirement	Comments
AM_Testing	ATE_COV.1 Evidence of coverage ATE_FUN.1 Functional testing ATE_IND.2 Independent testing – sample	Evidence of coverage of testing has been provided Functional testing has been performed by the developer The functional testing has been performed by an independent third party.
AM_Strength_of_function_ Evaluation	AVA_SOF.1 Strength of TOE security function evaluation	The strength of function analysis has been performed by the developer
AM_Vulnerability_analysis	AVA_VLA.1 Developer vulnerability analysis	The vulnerability analysis has been performed by the developer

One of the reasons for selecting the assurance measures identified here is to ensure that the TOE Security Functions cannot be tampered with or bypassed. All accesses to objects in the system are controlled, as is the flow of information, in accordance with the Diversinet access control policy and SPEx information flow policy, thus preventing the bypassing of the TSF. The TSF are protected from tampering by the same access control rules – only authorized security officers have the ability to modify the TSF, and they are assumed not to do that. The assurance mechanisms described above have been selected to help ensure that this is in fact true of the system as implemented.

6.3 Assurance Level

Based on the analysis provided in Sections 6.1 and 6.2, the Diversinet Passport Certificate Server v.4.1.1 meets the requirements identified in the Common Evaluation Methodology (CEM), Version 1.0, as EAL-2, augmented with an informal security policy model. That is, the TOE addresses all of the requirements of EAL-2, and additionally meets ADV_SPM.1 by providing an informal model of the system security policies.

Diversinet has chosen to implement the security assurance measures documented in Section 6.3 because extensive interaction with our customers has convinced us that this satisfies their needs in operating a PKI infrastructure. That is, the EAL-2 evaluated assurance level was a customer requirement.

The Diversinet Passport Certificate Server version 4.1.1 uses cryptographic functions to provide some of the security features. Thus, it was important that the system meet the FCS_CKM.1 requirement. As noted in Section 5.1.9 of this Security Target, FCS_CKM.1 has a dependency on FMT_MSA.2, and that requirement requires the assurance measure ADV_SPM.1, Informal Security Policy Model. Diversinet developed an informal model of the security policy enforced by the Passport Certificate Server v 4.1.1, as required by ADV_SPM.1. Thus, Diversinet has augmented the requirements of EAL-2 with the informal security policy model required by ADV_SPM.1.

7 TOE Summary Specification

7.1 Statement of TOE Security Functions

The TOE IT Security Functions and their specifications are listed as follows.

ITSF_ACF.1	The TOE performs access control based on user role, in accordance with The Diversinet Access Control Policy. This restricts the ability to change_default and modify certificates to the Security Officer role.
ITSF_ACF.2	The TOE provides the following roles: Security Officer, Administrator, and User. The Security Officer has the ability to disable and enable the function: creation of administrators. In addition, the TOE restricts the ability to disable and enable the function: user lock ID to the user and administrator roles. The TOE restricts the ability to change the user public key and user private key to the user. The TOE restricts the ability to override default values of information in certificates to the Administrator. .
ITSF_AUD	The TOE generates a log file that records the following audit events: a) Start-up and shutdown of the audit functions b) All auditable events for the minimum level of audit; and c) Publication of Certificates, Assignment of Alias, Deletion of all Retired Certificates, Get First ID Information, Get ID Information by Alias, Get Next ID Information, Get Previous ID Information, Lock ID, Set FA, Get Last ID Information, Delete Retired Certificate, Delete ID, Get First ID Information by FA, Get ID Information by ID, Get Next ID Information by FA, Get Previous Next ID Information by FA, Reserve ID, Unlock ID, Get Last ID Information by FA.
ITSF_CERT.1	The TOE creates certificates that contain the client ID, realm ID and public key of the associated user, signed by the CA (Server)
ITSF_CERT.2	The TOE generates evidence of validity and integrity for certificates in the form of an RSA-compliant digital signature.
ITSF_CERT.3	The TOE causes cryptographic keys to be generated in accordance with the identify current algorithm using a non-standard key generation method by calling a cryptographic module, outside the TOE boundary.
ITSF_CERT.4	The TOE publishes certificates in an Oracle database, outside the TOE boundary
ITSF_CERT.5	The TOE performs access of a user's public key by the RSA signature verification
ITSF_CERT.6	The TOE destroys users' private keys (stored in an encrypted state) by deletion from the Windows NT Registry
ITSF_CERT.7	The TOE can export certificates and encrypted private keys.
ITSF_CERT.8	The TOE can import certificates under user control, requiring the password for user's private key

ITSF_OTS	The TOE verifies that a one-time secret (OTS) satisfies the condition *that it satisfies the minimum length requirements* and has a Strength of Function rating of SOF-Basic
ITSF_SPEX	The TOE provides a secure transmission protocol, SPEx that ensures the confidentiality, integrity, replay-protection and completeness of all transmissions between servers.
ITSF_TS	The TOE creates, maintains and uses reliable time stamps for internal processing.
ITSF_UA.1	TOE requires user ID to be authenticated prior to any other actions involving encryption/decryption access of protected data.
ITSF_UA.2	TOE provides only asterisks (*) display as user inputs characters of authentication string
ITSF_UID.1	User ID is required by TOE at Login prior to any other actions involving encryption/decryption access of protected data.
ITSF_UID.2	The TOE has log-in verification by passphrase enforcing the following rules: a) Any user / non-user may fetch a certificate; b) A user may change his/her own key pair and may lock his/her user ID; c) An administrator may change all other IDs through access t the certificate database; d) The Security Officer must identify himself/herself by password. e) The passphrase mechanism must have a Strength of Function rating of SOF-Medium.

8 Protection Profile Claims

No protection profile claims are made in this security target.

9 Rationale

9.1 Security Objectives Rationale and Traceability

The purpose of this section is to show that the security objectives of the TOE are appropriate to the security problem defined in the security environment section. This is accomplished by showing that the security objectives adequately address the threats and assumptions defined in that security environment section. According to the Common Evaluation Methodology (CEM) paragraph 352, all security objectives for the TOE must be traced back to aspects of the identified threats to be countered and/or aspects of the organizational security policies to be met by the TOE. According to CEM paragraph 354, all security objectives for the environment must be traced back to aspects of the identified threats to be countered by the TOE's environment and/or aspects of the organizational security policies to be met by the TOE's environment and/or assumptions to be met in the TOE's environment. (Since there are no organizational security policies in this ST, the first requirement reduces to a requirement to map the TOE security objectives to the threats, and the second requirement reduces to a requirement mapping each environmental security objective to an assumption or a threat.)

To fulfill those requirements, we present below two tables that map security objectives to threats and assumptions. The first table maps TOE security objectives to threats. The second table maps environmental security objectives to threats and assumptions. For security objectives that map to threats, an informal argument is provided to show, for each security objective, why the identified security objective provides an effective countermeasure to the identified threat or mitigates risk to acceptable level. For security objectives that map to assumptions, an informal argument is given to show that meeting that assumption will satisfy that particular security objective.

9.1.1 Mapping of TOE Security Objectives to Threats

TOE Security Objective	Threat	Rationale
O.AccessControl -	T.MASQUERADE T.IMPORT	The TOE must control access, restricting access to authorized users when installed correctly and administered by a security officer and administrator. The TOE must also ensure that all users are accountable for their actions, and that each action can be traced to the user who caused it to happen. The TOE must provide a protection against unauthorized importation of certificates to the server certificate database. The digital signature of the server must distinguish authentic certificates from invalid certificates.

TOE Security Objective	Threat	Rationale
O.Accountability	T.MASQUERADE	The TOE must control access, restricting access to authorized users when installed correctly and administered by a security officer and administrator. The TOE must also ensure that all users are accountable for their actions, and that each action can be traced to the user who caused it to happen.
O.Import	T.IMPORT	The TOE must provide a protection against unauthorized importation of certificates to the server certificate database. The digital signature of the server must distinguish authentic certificates from invalid certificates.
O.NonRepudiation	T.REPUDIATION	The TOE must provide a mechanism to authenticate third party application transactions and services through secure public key distribution
O.Replay	T.REPLAY	The TOE must be able to detect when a communication constitutes a replay of a previous successful communication
O.CERT_VALS.CERT_CHANGE	T.MASQUERADE	The TOE must control access, restricting access to authorized users when installed correctly and administered by a security officer and administrator. The TOE must also ensure that all users are accountable for their actions, and that each action can be traced to the user who caused it to happen.

TOE Security Objective	Threat	Rationale
O.CERT_VALS.DEFAULT_OVERRIDE	T.MASQUERADE	The TOE must control access, restricting access to authorized users when installed correctly and administered by a security officer and administrator. The TOE must also ensure that all users are accountable for their actions, and that each action can be traced to the user who caused it to happen.
O.CERT_VALS.RESTRICT_CHANGE	T.MASQUERADE	The TOE must control access, restricting access to authorized users when installed correctly and administered by a security officer and administrator. The TOE must also ensure that all users are accountable for their actions, and that each action can be traced to the user who caused it to happen.
O.CERT_VALS.SEC_VALS	T.MASQUERADE	The TOE must control access, restricting access to authorized users when installed correctly and administered by a security officer and administrator. The TOE must also ensure that all users are accountable for their actions, and that each action can be traced to the user who caused it to happen.
O.KeyManagement	T.REPUDIATION	The TOE must provide a mechanism to authenticate third party application transactions and services through secure public key distribution

TOE Security Objective	Threat	Rationale
O.Key_Distr	T.REPUDIATION	The TOE must provide a mechanism to authenticate third party application transactions and services through secure public key distribution
O.KeyAccess	T.REPUDIATION	The TOE must provide a mechanism to authenticate third party application transactions and services through secure public key distribution
O.Roles.ID	T.ROLES	The TOE must provide roles that require hierarchical approval and control for the creation of roles of lower privilege.
O.Roles.Create	T.ROLES	The TOE must provide roles that require hierarchical approval and control for the creation of roles of lower privilege.
O.OneTimeSecret	T.REPUDIATION	The TOE must provide a mechanism to authenticate third party application transactions and services through secure public key distribution
O.TimeStamp	T.REPLAY	The TOE must eliminate the potential for replay attacks over internal transmissions of sensitive information by the use of a protocol in which digitally signed time stamps and integrity-checking is enforced.
O.CertConfidentiality	T.EAVESDROP	The TOE must prevent unauthorized disclosure of transmitted sensitive information between the server and client.

TOE Security Objective	Threat	Rationale
O.Cert_Integ	T.REPUDIATION	The TOE must provide a mechanism to authenticate third party application transactions and services through secure public key distribution
O.Trans_conf	T.EAVESDROP	The TOE must prevent unauthorized disclosure of transmitted sensitive information between the server and client.
O.TRANS_INTEG.SPEX	T.REPLAY	The TOE must eliminate the potential for replay attacks over internal transmissions of sensitive information by the use of a protocol in which digitally signed time stamps and integrity-checking is enforced.
O.UserID	T.MASQUERADE	The TOE must control access, restricting access to authorized users when installed correctly and administered by a security officer and administrator. The TOE must also ensure that all users are accountable for their actions, and that each action can be traced to the user who caused it to happen.
O.UserAuth	T.MASQUERADE	The TOE must control access, restricting access to authorized users when installed correctly and administered by a security officer and administrator. The TOE must also ensure that all users are accountable for their actions, and that each action can be traced to the user who caused it to happen.

TOE Security Objective	Threat	Rationale
O.AuthFeedback	T.MASQUERADE	The TOE must protect the passphrase value being entered by the Security Officer from direct or indirect disclosure. If an attacker can see (directly or indirectly) the passphrase being entered, the attacker can potentially impersonate the Security Officer at a later time.

9.1.2 Mapping of Environmental Security Objectives to Threats and Assumptions

Environmental Security Objective	Threat or Assumption	Rationale
O.CryptoKeyProtection	T.KeyTheft	The TOE must protect private keys from disclosure to or modification by unauthorized users, else it will be possible for attackers who gain access to the private keys to defeat the security provided by the system.
O.NO_EVIL	A.NO_EVIL.ADMIN	The TOE must be administered by trusted individuals who demonstrate knowledge of and adherence to the organizational security policy and procedures.
O.NO_OBSERV	A.NO_OBSERVATION	The TOE must be operable in an immediate environment of relative privacy with respect to other users, so that direct observation of Security Officer and Administrator login passwords / passphrases is not possible. Additionally, only asterisks are displayed on the screen while the passphrase is being entered, so that anyone who can indirectly observe the screen cannot see the characters being typed.

Environmental Security Objective	Threat or Assumption	Rationale
O.PHYS_ACC	A.SECUREAREA	The TOE must be operated in a physically secure environment, so that unauthorized access to and/or modification of the TOE hardware and software is prevented.
O.PWD_SHARE	A.NOSHARING	Users must not share passwords, because doing so would result in giving away private keys.
O.SEC_AWARE	A.NOSHARING A.NO_EVIL.ADMIN	All system users – including those who fill the roles of Security Officer, Administrator, and User – must be aware of and enforce the organization's security policy.
O.UNATTEND	T.MASQUERADE	The Organization must inform and train users in the proper procedures for unattended sessions in which sensitive information is accessible on their PC / workstations
O.SEC_LINK	A.ORACLE	The Oracle database is used as a repository for certificates and related information. It must always be configured correctly and available to the Certificate Server so that new certificates can be created and stored and existing certificates can be returned to the user.

9.2 Security Functional Requirements Rationale

9.2.1 Security Functional Requirements (SFRs) Rationale

The rationale for the SFRs against the security objectives of the TOE is given in the table below. For each security objective of the TOE, a list of assigned SFRs is given, followed by an argument stating how each SFR addresses or satisfies the security objective in question.

Security Objective	SFR	Rationale

Security Objective	SFR	Rationale
O.AccessControl	FDP_ACC.2	FDP_ACC.2 enforces the Diversinet Access Control Policy on the specific subjects and objects as required by the TOE access control policy
O.AccessControl	FDP_ACF.1	FDP_ACF.1 enforces the Diversinet Access Control Policy pertaining to specific objects and the relations among those objects as required by the TOE access control policy
O.AccessControl	FDP_ETC.2	FDP_ETC.2 provides that the TSF must enforce the Diversinet Access Control Policy when exporting certificates and encrypted private keys.
O.CERT_INTEG	FDP_SDI.1	FDP_SDI.1 provides that the TSF monitor the user certificates for integrity based on the digital signature of the certificate.
O.TRANS_CONF	FDP_UCT.1	FDP_UCT.1 provides that the TSF shall protect sensitive certificate and key objects from unauthorized disclosure during transmission.
O.OneTimeSecret	FIA_SOS.1	FIA_SOS.1 provide that the quality metrics of size and uniqueness of one-time secrets be enforced, and that the strength of function requirement of SOF-Basic (for One-Time Secrets) is met.
O.ROLES.ID	FMT_SMR.2	FMT_SMR.2 provides that the TSF maintain the roles: Security Officer, Administrator and User, and that it shall be able to associate users with those roles.
O.ROLES.CREATE	FMT_MOF.1	FMT_MOF.1 provides that the TSF restrict the ability to create and remove Administrators to the Security Officer role. Furthermore, the ability to create and remove users is restricted to the Administrator role.
O.CERT_VALS.CERT_CHANGE	FMT_MSA.1	FMT_MSA.1 provides that the TSF restrictions which have the ability to change_default, query and modify be present.
O.CERT_VALS.DEFAULT_OVERRIDE	FMT_MSA.3	FMT_MSA.3 provides that the TSF allows only the Administrator role to specify alternative initial values to override the default values on creation of a certificate.
O.CERT_VALS.RESTRICT_CHANGE	FMT_MTD.1	FMT_MTD.1 provides that the TSF restrict the ability to change user attributes in the certificate to the Administrator role.
O.CERT_VALS.SEC_VALS	FMT_MTD.3	FMT_MTD.3 provides that the TSF accept only secure values for user data in the certificate.

Security Objective	SFR	Rationale
O.TimeStamp	FPT_STM.1	FPT_STM.1 provides that the TSF use reliable internal time stamps to support certificate management functions.
O.USER_ID	FIA_UID.2	FIA_UID.2 provides that identification (by selection of user ID from the login form) must be done prior to any other TSF actions, such as encryption / decryption of protected data.
O.USER_AUTH.	FIA_UAU.2	FIA_UAU.2 provides that the user cannot perform actions such as encryption / decryption of protected data (other than selection of user ID from the login form) prior to authentication of the user's identity.
O.AUTH_FEEDBACK	FIA_UAU.7	FIA_UAU.7 provides that the authentication feedback to the user be limited, allowing specifically that the characters of the user authentication string could be represented by asterisks (*).
O.Accountability	FAU_GEN.1	FAU_GEN.1 covers the generation of audit records, which are an important part of meeting the overall Accountability security objective.
O.Accountability	FAU_SAR.1	FAU_SAR.1 ensures that audit records can be reviewed by a competent Security Officer, thus ensuring that the accountability security objective can be met.
O.NonRepudiation	FCO_NRO.1	FCO_NRO.1 provides that the system be able to determine the originator of a valid transmitted certificate, which satisfies the security objective.
O.KeyManagement	FCS_CKM.1	FCS_CKM.1 controls the generation of cryptographic keys, which is a part of the key management security objective.
O.KeyManagement	FCS_CKM.4	FCS_CKM.4 controls the destruction of cryptographic keys, which is a part of the key management security objective.
O.KeyManagement	FCS_COP.1	FCS_COP.1 covers the usage of cryptographic keys in system operation, which is a part of the key management security objective.
O.Cert_Integ	FDP_DAU.1	FDP_DAU.1 ensures that the contents of a certificate have not been modified in an unauthorized manner.
O.TRANS_INTEG.SPEX	FDP_IFC.1	FDP_IFC.1 and FDP_IFF.1 enforce the basic SPEx information flow control policy, meeting this security objective.

Security Objective	SFR	Rationale
O.TRANS_INTEG.SPEX	FDP_IFF.1	FDP_IFC.1 and FDP_IFF.1 enforce the basic SPEx information flow control policy, meeting this security objective.
O.Accountability	FIA_ATD.1	FIA_ATD.1 covers the specific information that must be known about a user, in order to ensure that accountability can be achieved.
O.CERT_VALS.SEC_VALS	FMT_MSA.2	FMT_MSA.2 ensures that only secure values are accepted and used in certificates.

The coverage of the above table against the SFRs satisfies the following properties:

- for every security objective of the TOE, there is at least one SFR that satisfies it;
- for every SFR, there is at least one security objective of the TOE that it addresses.
- for every security objective of the TOE, an informal argument as to why the identified SFRs are sufficient to meet it is provided.

9.2.2 Functional Claims Rationale

The selected functionality for this ST is consistent with and appropriate for the security objectives for the TOE. There are three main categories of security service that the TOE provides:

- All authentication is through public key encryption using specified algorithms and key lengths that are of appropriate strength for business, financial and personal private data under a broad class of applications. The key component of this service is the public key certificate that authenticates a user's public key;
- User Identification and Authentication of Security Officers and Administrators must precede all other access to protected information, providing binding between the privileged role and the private and symmetric encryption keys used in read and write access to protected information stores;
- Protection of private keys in storage on the client Windows NT registry is by symmetric encryption.

These security services embody the security objectives of the TOE and are consistent with the level of capability and motivation that a threat agent would be expected to possess, given the assumptions regarding data sensitivity of information assets and sophistication of threat agent. Elimination of all potential threat agents clearly requires environmental support, procedural security and training. The latter safeguards are complementary security objectives that the environment is expected to supplement the TOE functional properties with in order to obtain an overall acceptable level of risk. They do not constitute weaknesses or omissions in the TOE, as the majority of the environmental security objectives are beyond the scope of any conceivable software solution. In addition, not all may represent serious risk to the average system in which the TOE is deployed.

9.2.3 Strength of Function Rationale

In keeping with policies adopted under the Common Evaluation Methodology and the Canadian Common Criteria Evaluation Scheme, strength of function claims are not made for security mechanisms based on cryptographic mechanisms. In addition, some mechanisms (e.g., access control functions) do not lend themselves to analysis of strengths. Therefore, Strength of Function claims are made in this ST for only two mechanisms:

- Keylock passphrases, used by the Security Officer to authenticate himself/herself to the Certificate Server application; and
- One-Time Secrets, used by clients to authenticate themselves to the Certificate Server upon first requesting the issuance ("publishing") of a certificate.

Keylock passphrases, called passphrases for short, are strings entered by the Security Officer during system installation. They are generated by the Security Officer through means not controlled by the Passport Certificate Server. The Passport Certificate Server checks that the chosen passphrase is sufficiently long, and then accepts or rejects it.

One-Time Secrets are strings entered at a client or Registration Authority (RA) device when the user first registers for a certificate, or when the client or RA needs to obtain a new Root Key Set. As with passphrases, the actual generation of the One-Time Secrets is not controlled by the Passport Certificate Server. However, the Passport Certificate Server v 4.1.1 User's Guide [UG] recommends that One-Time Secrets be between 8 and 48 characters long for adequate security.

Both passphrases and One-Time Secrets are case-sensitive, and can include letters, digits, and certain special characters.

The strength of function claim made for passphrases is SOF-Medium, and for One-Time Secrets is SOF-Basic. These strengths are appropriate for the security objectives O.OneTimeSecret, O.USER_ID, and O.USER_AUTH, and are consistent with other mechanism and assurance levels.

9.3 TOE Summary Specification Rationale

9.3.1 IT Security Functions Rationale (SFRs)

The TOE IT Security Functions are listed with cross-references to the SFRs, described in Section 5.1, that are provided by the defined IT Security Function. Specifications of IT Security Functions are provided in section 7.1. A Coverage Mapping is included to describe how the IT Security Function covers the referenced SFRs.

Security Functional Requirement	IT Security Function(s)	IT Security Function to SFR Coverage Mapping

Security Functional Requirement	IT Security Function(s)	IT Security Function to SFR Coverage Mapping
FAU_GEN.1	ITSF_AUD	It is required that an audit record of the following auditable events be supported by the TSF: a) Start-up and shutdown of the audit functions; b) All auditable events for the minimum level of audit; and c) Publication of Certificates, Assignment of Alias, Deletion of all Retired Certificates, Get First ID Information, Get ID Information by Alias, Get Next ID Information, Get Previous ID Information, Lock ID, Set FA, Get Last ID Information, Delete Retired Certificate, Delete ID, Get First ID Information by FA, Get ID Information by ID, Get Next ID Information by FA, Get Previous Next ID Information by FA, Reserve ID, Unlock ID, Get Last ID Information by FA. Each of the above actions is performed by the TSF through ITSF_AUD.
FAU_SAR.1	ITSF_AUD	It is required that the TSF shall provide Security Officers with the capability to read message type, event type from the audit records and further that The TSF shall provide the audit records in a manner suitable for the user to interpret the information. This is performed by the TSF through ITSF_AUD.
FCO_NRO.1	ITSF_CERT.1	It is required that the TSF shall be able to generate evidence of origin for certificates at the request of the recipient, and that the TSF shall be able to relate the client ID (contains issuer's ID), realm ID, public key, signature algorithm(s) of the originator of the information, and the certificate serial ID, sequence identifier, client ID (contains issuer's ID), realm ID, functional attribute, public key, signature algorithm(s) of the information to which the evidence applies. Furthermore, it is required that the TSF shall provide a capability to verify the evidence of origin of information to recipient given no limitations on the evidence of origin. Each of the above requirements is performed in the TSF through ITSF_CERT.1.
FCS_CKM.1	ITSF_CERT.3	It is required that the TSF ensure that 1024-bit RSA keys are generated in accordance with RSA PKCS #1 and FIPS 186-2. ITSF_CERT.3 ensures that, using the BSAFE toolkit, these keys are properly generated.
FCS_CKM.4	ITSF_CERT.6	It is required that the TSF ensure the secure deletion of cryptographic keys. The TSF satisfies this requirement through ITSF_CERT.6
FCS_COP.1	ITSF_CERT.5	It is required that the TSF ensure that RSA keys are properly used to encrypt, decrypt, sign and verify signatures on information in accordance with RSA PKCS#1 and FIPS 186-2. ITSF_CERT.5 ensures that, using the BSAFE toolkit, these operations are correctly performed.
FDP_ACC.2	ITSF_ACF.1	It is required that the TSF shall enforce the Diversinet Access Control Policy on Subjects: Security Officer, Administrator and User, and objects: certificates and all operations among subjects and objects covered by The Diversinet Access Control Policy. Furthermore it is required that all operations between any subject in the TSC and any object within the TSC are covered by an access control SFP. The TSF satisfies this requirement through ITSF_ACF.1 which enforces The Diversinet Access Control Policy.

Security Functional Requirement	IT Security Function(s)	IT Security Function to SFR Coverage Mapping
FDP_ACF.1	ITSF_ACF.2	It is required that the TSF shall enforce the Diversinet Access Control Policy on objects based on type of object. Furthermore, the TSF must enforce the following rules to determine if an operation among controlled objects is allowed: a) the private key of the appropriate CA must be used to sign user certificates b) the public key assigned to the User must be contained in his/her certificate. The TSF must also explicitly deny access of subject to objects based on the rules: a) The private key of a user cannot be accessed by any other user. The TSF satisfies these requirements through ITSF_ACF.1, which enforces The Diversinet Access Control Policy.
FDP_DAU.1	ITSF_CERT.2	It is required that the TSF shall provide a capability to generate evidence that can be used as a guarantee of the validity of public key certificates. The TSF must also provide users with the ability to verify evidence of the validity of the public key certificates. The TSF provides this functionality through ITSF_CERT.2, which generates evidence of validity and integrity of certificates in the form of an ECDSA-compliant digital signature.
FDP_ETC.2	ITSF_CERT.2 ITSF_CERT.4 ITSF_ACF.1 ITSF_CERT.7	It is required that the TSF perform the following requirements: a) Enforce The Diversinet Access Control Policy when exporting user data, controlled under the SFPs, outside of the TSC b) Export the user data with the user data's associated security attributes. c) Ensure that the security attributes, when exported outside the TSC, are unambiguously associated with the exported user data. Requirement a) is satisfied by the TSF through ITSF_CERT.4 (when exporting certificates to the Oracle database) and ITSF_CERT.7 (when exporting certificates to other locations; e.g., to clients); b) and c) are satisfied through ITSF_ACF.1 and ITSF_CERT.2, which bind user security attributes in the certificate through digital signatures.
FDP_IFC.1.1	ITSF_ACF.1 ITSF_SPEX	It is required that the TSF enforce The SPEx Information Flow Policy on the following: a) A new client subject (in open mode) or an Administrator subject (in closed mode) may transmit a client-generated one-time secret (OTS) to the server subject encrypted with the server's public key (in open mode) b) The CS subject may generate a user ID and transmit this object to the new client subject (in open mode) or to an Administrator (Registration Authority in closed mode).
FDP_IFF.1	ITSF_SPEX	It is required that the TSF protect and control data stored and transmitted in the system according to a set of attributes. The TSF uses the SPEx Information Flow Policy and controls to meet that requirement.
FDP_SDI.1	ITSF_CERT.2	It is required that the TSF monitor user data stored within the TSC for integrity errors on all objects, based on the digital signature of certificate. The TSF satisfies these requirements through ITSF_CERT.2.

Security Functional Requirement	IT Security Function(s)	IT Security Function to SFR Coverage Mapping
FDP_UCT.1	ITSF_SPEX	It is required that the TSF enforce The SPEx Information Flow Policy to be able to transmit objects in a manner protected from unauthorized disclosure. The TSF satisfies this condition through The SPEx Information Flow Policy, by implementing encryption of sensitive information.
FIA_ATD.1	ITSF_CERT.1	It is required that the TSF maintain the following list of security attributes belonging to individual users: client ID (contains issuer's ID), realm ID, public key. The TSF satisfies these requirements through ITSF_CERT.1.
FIA_SOS.1	ITSF_OTS	It is required that the TSF shall provide a mechanism to generate secrets that meet the defined quality metrics: a) provides Strength of Function SOF-Basic for One-Time Secrets, and SOF-Medium for passphrases , and b) the OTS has never been used before. The TSF satisfies these requirements through ITSF_OTS.
FIA_UAU.2	ITSF_UA.1	It is required that the TSF ensure that each user is successfully authenticated before allowing any other TSF-mediated actions on behalf of that user. The TSF satisfies this requirement through ITSF_UA.1.
FIA_UAU.7	ITSF_UA.2	It is required that the TSF shall provide only asterisks to be displayed to the user while the authentication is in progress. The TSF satisfies this requirement through ITSF_UA.2.
FIA_UID.2	ITSF_UID.1	It is required that the TSF enforce each user to identify itself before allowing any other TSF-mediated actions on behalf of that user. The TSF satisfies this requirement through ITSF_UID.1
FMT_MOF.1	ITSF_ACF.2	It is required that the TSF restrict the ability to disable and enable the function: creation of administrators to the Security Officer roles. In addition, the TSF must restrict the ability to disable and enable the function user lock ID to the administrator role. These requirements are met by the TSF through ITSF_ACF.2.
FMT_MSA.1	ITSF_ACF.1	It is required that the TSF enforce the Diversinet Access Control Policy to restrict the ability to change_default and modify. These requirements are met by the TSF through ITSF_ACF.1.
FMT_MSA.2	ITFS_ACF.1 ITSF_ACF.2	It is required that the TSF ensure that only secure values are used for security attributes. The default values configured at installation time will be secure, given the assumption that administrators will correctly configure the system. ITSF_ACF.1 and ITSF_ACF.2 will then ensure that any changes to these defaults are also secure.

Security Functional Requirement	IT Security Function(s)	IT Security Function to SFR Coverage Mapping
FMT_MSA.3	ITSF_ACF.1 ITSF_ACF.2	It is required that the TSF enforce the Diversinet Access Control Policy to provide restrictive default values for security attributes that are used to enforce the SFP and that the Administrator specify alternative initial values to override the default values when an object or information is created. . The TSF ensures that the above policy is enforced through ITSF_ACF.1, thus ensuring that default values are enforced. The TSF further allows only the Administrator to override the above policy by specifying an alternate default value, through ITSF_ACF.2
FMT_MTD.1	ITSF_ACF.2	It is required that the TSF restrict the ability to change the user public key, user private key and certificate to the User role and that the ability to lock the user's ID be restricted to the User and the Administrator. It is further required that the TSF shall restrict the ability to unlock the user's ID to the Administrator. These requirements are met through ITSF_ACF.2
FMT_MTD.3	ITSF_ACF.2	It is required that the TSF shall ensure that only secure values are accepted for TSF data. The TSF ensures that default values as specified by the Administrator (which are the "secure values" for a given implementation) can only be changed by the Administrator. Additionally, the TSF ensures that only the user can change the public key.
FMT_SMR.2	ITSF_ACF.2 ITSF_UID.2	It is required that The TSF shall maintain the roles: Security Officer, Administrator, User, and be able to associate users with roles. The TSF ensures that this is met through ITSF_ACF.2 (regarding roles), and that association with users is performed through an administrative login process satisfying ITSF_UID.2
FPT_STM.1	ITSF_TS	It is required that the TSF provide reliable timestamps for its own use. The TSF satisfies this requirement through ITSF_TS.

The combined aggregate of the TOE security functions satisfy the set of identified TOE SFRs as shown above. Given that the cryptographic power of the TOE (in terms of algorithm choice and key size) is sufficient to protect information assets within the requirements of the organization / environment, then it can be concluded that the security functionality of the TOE is effective in applying that cryptographic protection to a restricted set of information assets.