Tokeneer ID Station
System Requirements Specification

S.P1229.41.1
Issue: 1.3
Status: Definitive
19th August 2008

Originator

David Cooper

Approver

Janet Barnes (Project Manager)

Copies to:

National Security Agency *Praxis High Integrity Systems*

SPRE Inc.

Contents

1 Introduction

In order to demonstrate that developing highly secure systems to the level of rigour required by the higher assurance levels of the Common Criteria is possible, the NSA has asked Praxis High Integrity Systems to undertake a research project to develop a high integrity variant of part of an existing secure system (the Tokeneer System) in accordance with Praxis' own high-integrity development process. This development work will then be used to show the security community that it is possible to develop secure systems rigorously in a cost-effective manner.

This document is the system requirements specification, explaining which aspects of Tokeneer will be re-developed and the requirements such a re-development will meet. It documents the results of the first step in the Praxis' high integrity systems development approach. The whole process consists of:

1 Requirements analysis (the REVEAL® process)

2 Formal specification (using the formal language Z)

3 Design (the INFORMED process)

4 Implementation in SPARK Ada

5 Verification (using the SPARK Examiner toolset)

2 Project Context

The development project for this high integrity variant has two main objectives:

1 to redevelop part of the software for the Identification Station (ID Station — part of the Tokeneer system) according to Praxis High Integrity System's formal, high-integrity system development process; and

2 to demonstrate that this rigorous development approach required by EAL5 can result in a more reliable product at lower cost when compared with traditional development methods.

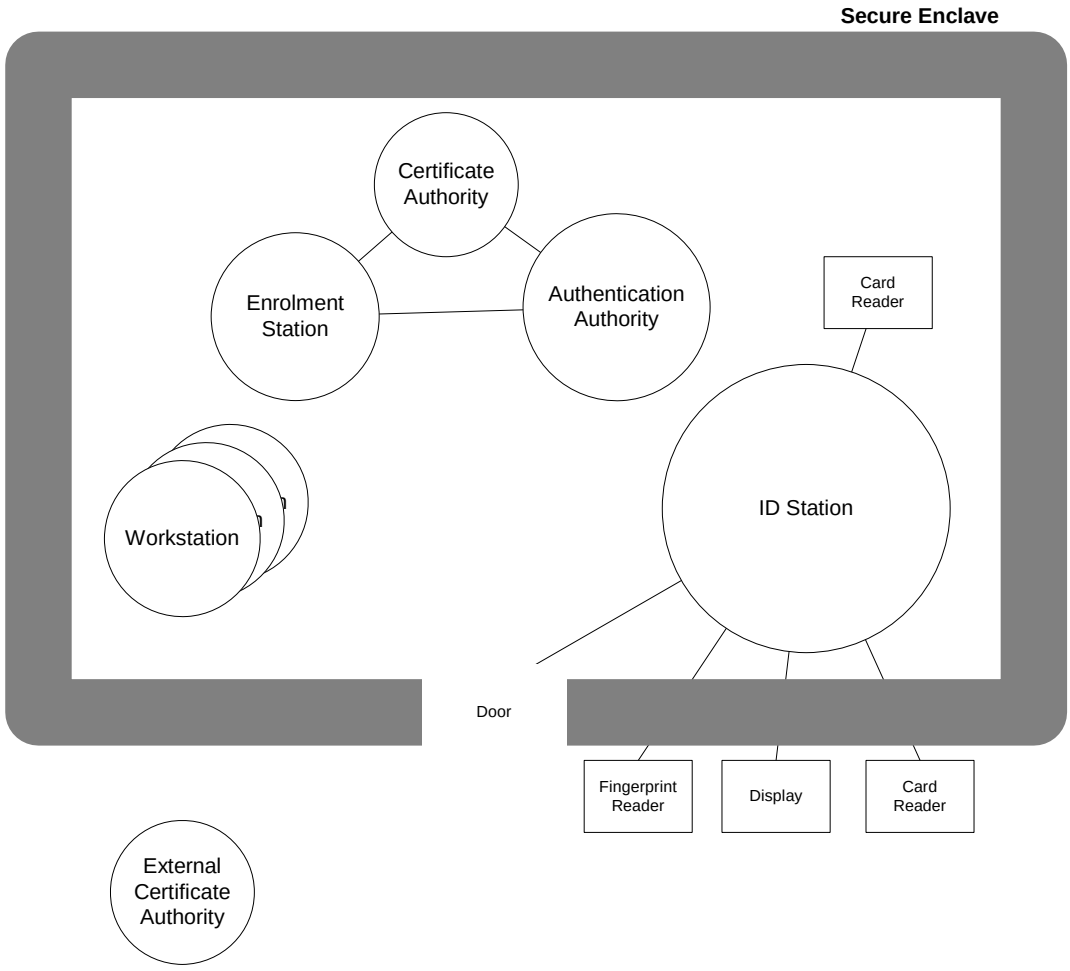
The ID Station high integrity variant development will produce or influence a number of "outputs": a set of metrics and process evidence (contained in the final summary report from the project); possible conference and journal publications and presentations; the actual software product itself (the re-developed ID Station software); and the demonstrated Praxis high-integrity development process used. These outputs will be the mechanisms that NSA will use to influence the other stakeholders, as follows.

Publications and conference presentations will be read and attended by members of Contractors and the NSA and will raise awareness of the findings. This will specifically include NSA-internal publications such as Next Wave. Specific metrics and evidence will be made available within NSA. This will encourage the NSA divisions to refer to the Praxis process in their calls for work from Contractors, and generally inform other parts of the NSA that the Praxis process is a desirable process to meet the high levels of the Common Criteria.

3 Domain Information

This section describes the existing Tokeneer system, and the existing ID Station. This allows the proposed variant system structure to be understood in context (see section 4).

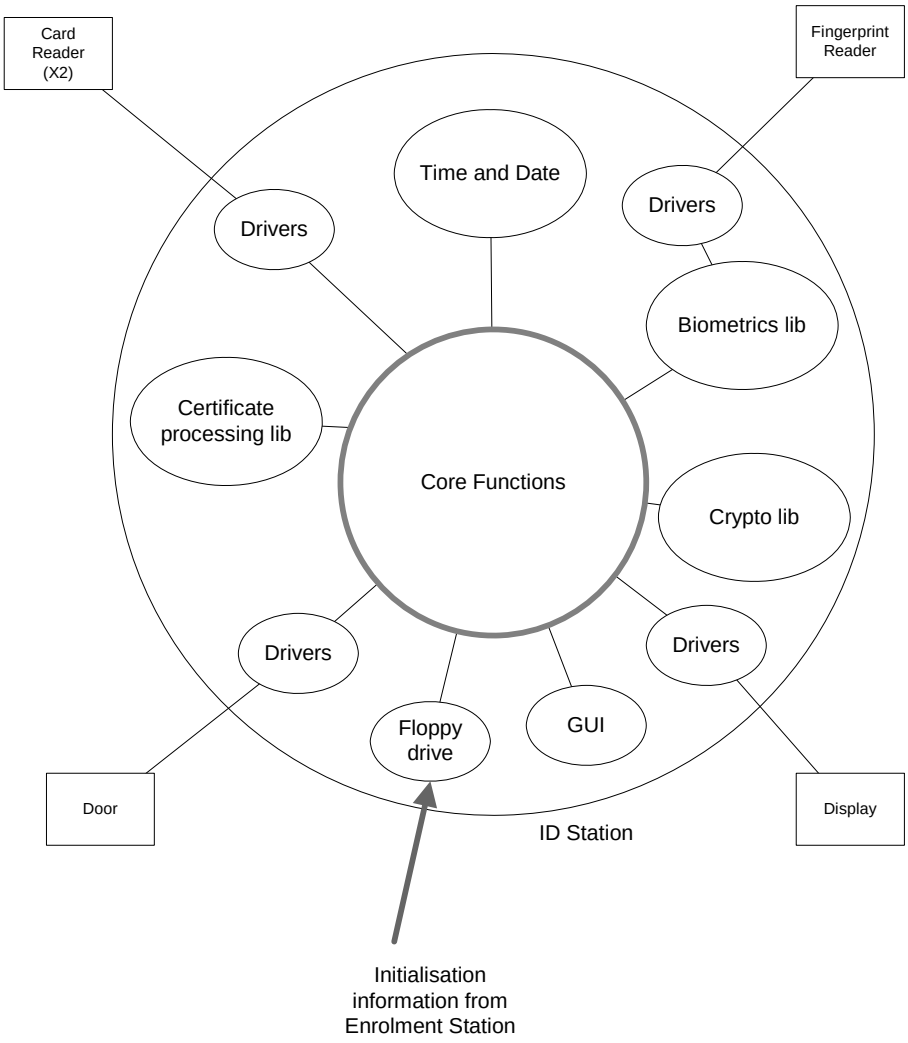
The ID Station is part of the larger Tokeneer system, as depicted below. *Analysis of the interactions within this larger system has not been carried out, and would normally be done to ensure that a full understanding of the system's true requirements is obtained. Within the scope of this project, and given that the functionality of the system is already well-defined, this step will not be carried out.*



Overall Tokeneer System:

The complete Tokeneer system consists of a secure enclave and a set of system components, some housed inside the enclave and some outside.

There are two system boundaries of interest in the development of this variant: the boundary between the ID Station machine and its environment (including its peripherals); and the boundary between the ID Station core functions and its support functions. These boundaries are expanded below.



Existing ID Station Structure:

The existing ID Station has four connected peripherals and a number of internal drivers and libraries.

The ID Station interfaces to four different physical devices:

Fingerprint reader
Smartcard reader (two instances)
Door
Visual display

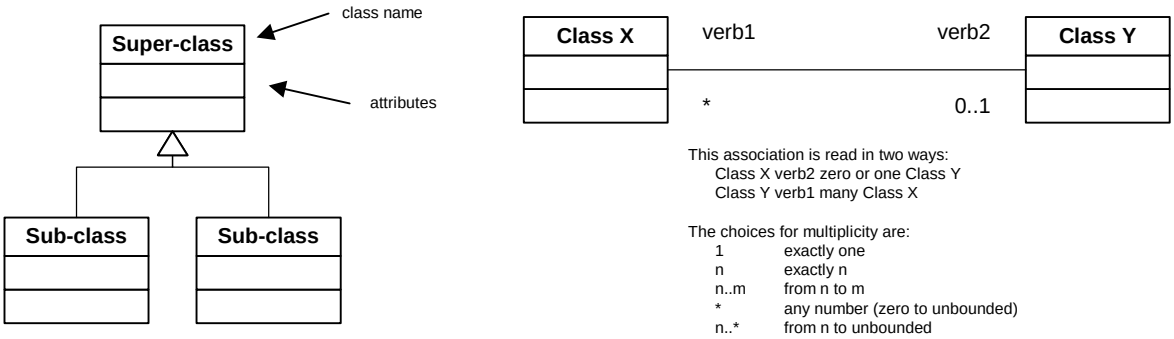
There is a fifth interface via the floppy drive, used to take initialisation data from the Enrolment Station and to take configuration data in, and audit data out. We take this to be a general-purpose interface to some read/write, removable media.

3.1 Certificates

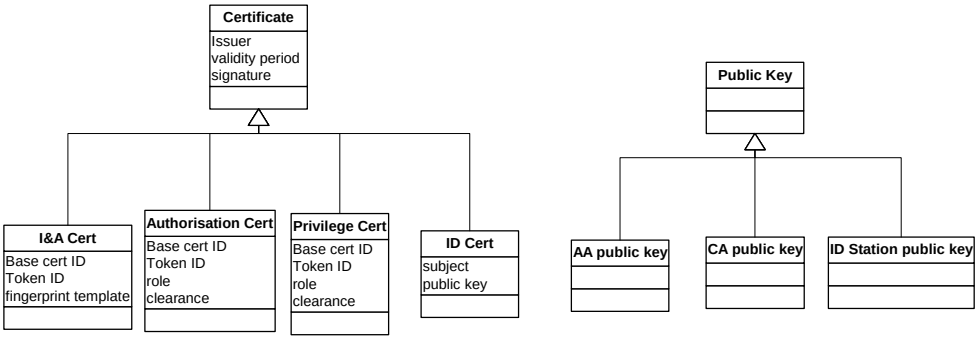
Types of certificates:

- ID Certificate
- I&A Certificate
- Privilege Certificate
- Authorisation Certificate

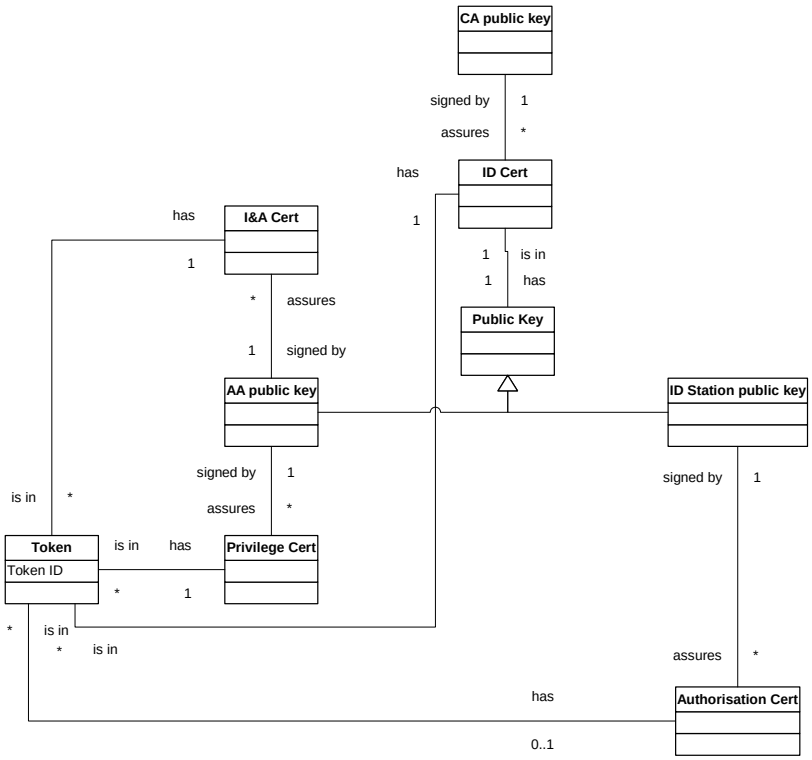
The relationships between the different types of certificate is shown using the UML Class Diagrams below:



UML Class Diagram key



Class hierarchy for certificates



Certificate relationships

3.2 Biometrics

All of the complexity of the biometrics is hidden within the biometrics library, which we will be simulating in a very simple way, enabling the test drivers to decide whether a fingerprint will or will not match a template.

3.3 Door/Latch

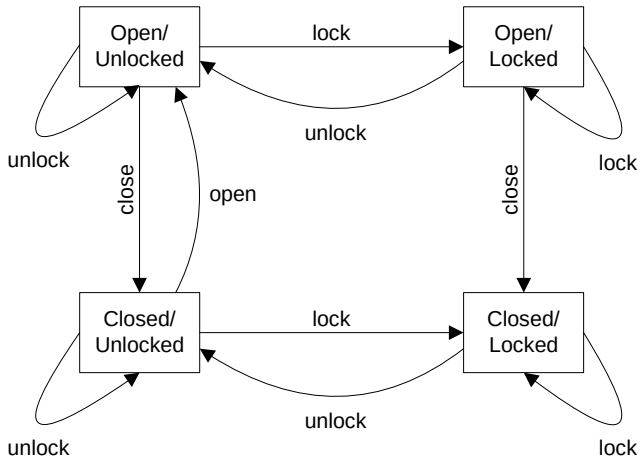
The door has four possible states: the cross-product of open/closed and locked/unlocked.

Open means the door does not prevent a human from entering or leaving the enclave.

Closed means the door prevents a human from entering or leaving the enclave. To enter or leave, the door must first be *opened*.

Locked means that if the door is closed, it cannot be opened. If the door is open, it can be closed.

Unlocked means that if the door is closed, it can be opened. If the door is open, it can be closed.

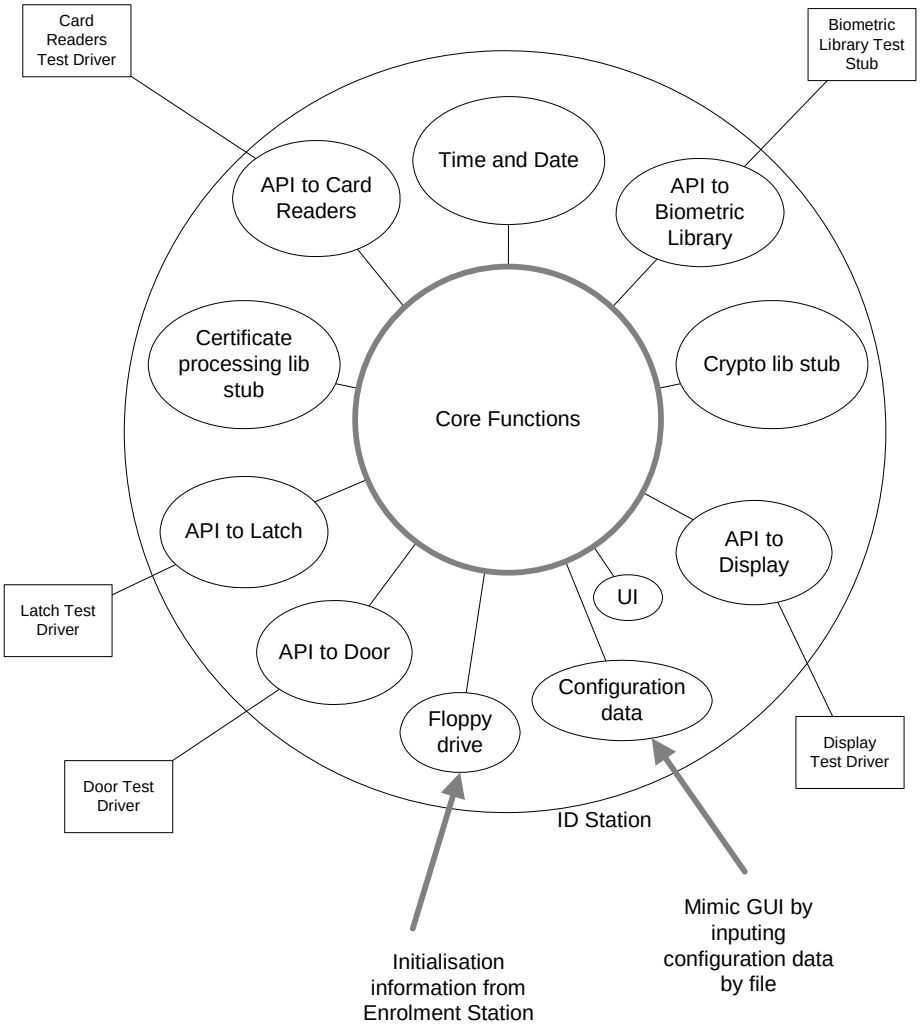


Open/Closed and Locked/Unlocked relationships

4 System Context

This section defines the system boundary for the high integrity variant of the ID Station, and explains which elements will be developed by whom, and by what process.

Taking the existing ID Station model given in section 3, we can re-present it showing how the elements will be re-developed.



High integrity variant ID Station structure

The peripherals are replaced by Test Drivers; the drivers by message translation mechanisms; and the libraries by stubs. The GUI is mimicked by a file, and simple command line interaction.

Praxis will develop the Core Functions according to their high-integrity development process. This is the part of the development that will be assessed for suitability for high assurance Common Criteria requirements.

Praxis will develop the modules within the ID Station but outside the Core Functions (except for Time and Date, which will be operating system supplied) in a sound and professional manner, but not necessarily according to a high-integrity process.

Peripheral Test Drivers (i.e. Card Reader Test Driver, Fingerprint Reader Test Driver, Door Test Driver, Latch Test Driver, Display Test Driver) will be developed by SPRE on a separate machine.

5 Behavioural Requirements

The required behaviour of the ID Station is specified here using scenarios, which run through typical uses of the ID Station and define the interaction between the ID Station and its connected systems. In each case the scenario focuses on a *successful* outcome, but it also covers various conditions that may arise that do not allow the successful outcome to be achieved. The full behaviour of the system, including both successful and failed outcomes, constitute the *system requirements.*

The scenarios considered are:

1 User gains allowed initial access to Enclave

2 User is denied prohibited initial access to Enclave

3 User gains allowed repeat access to Enclave

4 ID Station is started and enrolled with input from the Enrolment Station

5 ID Station is started already enrolled

6 ID Station is shut down

7 Security Officer updates the configuration of the ID Station

8 Audit log is archived

9 Guard manually unlocks the door

10 Administrator logs on

11 Administrator logs off

5.1 General issues

5.1.1 Failure conditions

Unless otherwise specified, a failure condition results in

ScGeneral.Fail.Locked
The door latch is locked.

ScGeneral.Fail.NoChange
There is no change to the values stored on the card inserted into the card reader, the configuration data or start-up data of the ID Station.

ScGeneral.Fail.Audit
The sequence of auditable events up to and including the failure condition have been recorded in the Audit Log.

5.1.2 Auditable events

The following events will be recorded in the Audit Log whenever they occur.

5.1.2.1 Start-up

1 System start-up

2 System shut-down

5.1.2.2 Administration

1 Log-on by Administrator

2 Log-off by Administrator

3 Modification of ID Station configuration data values

4 Clearing audit log

5 Creating an audit log archive file

6 Writing audit log archive file to floppy

7 Confirmation of successful write to floppy (by read-back)

8 Expiration of any timeout

9 Invocation of any command from the console: change configuration, archive audit log, unlock door, shutdown

5.1.2.3 Use

1 Insertion of card

2 Removal of card

3 Reading data from card (success or failure)

4 Writing data to the card (success or failure)

5 Reading fingerprint image

6 Setting the latch to locked.

7 Setting the latch to unlocked.

8 Recognising the door has opened.

9 Recognising the door has closed.

10 Writing data to the display.

11 Validation of any certificate (success or failure)

12 Creation or modification of signed certificates

13 Comparison of fingerprint image and template (success or failure)

14 Change of alarm state (to alarming or to silent)

15 Change of audit alarm state (to alarming or to silent)

5.2 Scenarios

5.2.1 User gains allowed initial access to Enclave

Description

A User who should be allowed access to the enclave is given access, making use of biometric authentication.

Stimulus

User inserts a smartcard into the smartcard reader.

Assumptions

ScGainInitial.Ass.ValidStart
The ID Station has valid start-up data.

ScGainInitial.Ass.ValidConfig
The ID Station has a valid data configuration.

ScGainInitial.Ass.Quiescent
The ID Station is quiescent (no other access attempts, configuration changes or start-up activities are in progress).

ScGainInitial.Ass.Secure
The User is outside the enclave; the door is closed and locked.

ScGainInitial.Ass.ValidUser
The card inserted by the User has a valid ID Certificate, I&A Certificate, and Privilege Certificate, **and** the card inserted by the User has a valid fingerprint template that matches the fingerprint of the User's finger.

ScGainInitial.Ass.PoorAC
The card inserted by the User does not have a valid, current Authorisation Certificate.

Success End-conditions

ScGainInitial.Suc.UserCard
The User has possession of the card he originally inserted.

ScGainInitial.Suc.GoodAC
The card inserted by the User contains a current, valid Authorisation Certificate with

- validity time: from now until now+(length of time specified in ID Station configuration data)
- security level: equal to the minimum of (the security level defined in the ID Station configuration data) and (the security level in the Permission Certificate on the card inserted by the User)

ScGainInitial.Suc.PersistCerts
The card inserted by the User contains the same, unchanged ID Certificate, I&A Certificate, and Privilege Certificate it had at the beginning of the scenario.

ScGainInitial.Suc.UserIn
The User is in the Enclave.

ScGainInitial.Suc.Locked
The Enclave door is closed and locked.

ScGainInitial.Suc.Audit
The following events have been recorded in the Audit Log (in any order), and the existing audit records are preserved:

- Insertion of card
- Removal of card
- Reading data from card (possibly multiple failures, but at least one success)
- Writing data to the card (possibly multiple failures, but at least one success)
- Reading fingerprint image

- Setting the door to locked.
- Setting the door to unlocked.
- Door opening
- Door closing
- Writing data to the display.
- Validation of any certificate (possibly multiple failures, but at least one success)
- Creation or modification of signed Authorisation certificate
- Comparison of fingerprint image and template (possibly multiple failures, but at least one success)

Failure Conditions

ScGainInitial.Fail.ReadCard
The card inserted by the User does not allow all its data to be successfully read, possibly due to being incorrectly inserted in the first place; being a faulty card; having the incorrect information on it; or being removed before all the information has been read. The set of data to be read is at least:

- ID Certificate
- I&A Certificate
- Privilege Certificate
- Fingerprint Template (contained in the I&A Certificate)

ScGainInitial.Fail.Fingerprint
A matching fingerprint has not been read, possibly due to no finger being presented to the fingerprint reader within X seconds of the display requesting a fingerprint; or the fingerprint not being successfully read within X seconds of the display requesting a fingerprint; or the fingerprint that was successfully read not being successfully matched to the template read from the card. The value X shall be taken from configuration data of the ID Station.

ScGainInitial.Fail.WriteCard
The card originally inserted by the User does not allow a new Authorisation Certificate to be successfully written, possibly due to being incorrectly inserted in the first place; being a faulty card; or being removed before all the information has been written.

ScGainInitial.Fail.UserSlow
The User is too slow in opening the door, so the door locks with the user still outside the enclave. Or the user opens the door, but chooses not to pass through, closing the door again.

ScGainInitial.Fail.DoorPropped
Once the door has been opened, it is not allowed to close (it is propped open).

ScGainInitial.Fail.Audit
Audit files cannot be successfully written. Result: the Door is locked and the system is shutdown.

ScGainInitial.Fail.AuditPreserve
Space for audit files has been exhausted. Result: the oldest audit records are overwritten with the new audit records, and an alarm is raised to the Guard.

Constraints

ScGainInitial.Con.NoInterleave
No ID Station restart or Configuration data changes will be allowed during this scenario.

Rationale

ScGainInitial.Ass.ValidUser and ScGainInitial.Ass.PoorAC
These assumptions are negated in the next scenario, *User is denied prohibited initial access to Enclave*, but we need system processing to be able to distinguish between them. This will lead to additional system functional specification.

ScGainInitial.Suc.PersistCerts
Note that by requiring the certificates to be preserved, all their constituent parts, such as the Fingerprint Template, will be preserved.

Issue

What is the value of “X” in ScGainInitial.Fail.Fingerprint?

ScGainInitial.Suc.GoodAC, ScGainInitial.Suc.PersistCerts and ScGainInitial.Fail.WriteCard
There may be an unacceptable performance implication of meeting these requirements. In order to be sure that the card inserted by the user has had the Auth Cert written to it, once the Auth Cert is written the ID Station will need to read all the information off again and compare it with the values originally read. Otherwise, someone could swap a different card into the card reader while the fingerprint was being taken. Without being able to monitor the state of the card reader (to detect the card being removed and re-inserted) this check will be needed. We could accept the system writing the Auth Cert to a different card and still allowing the enclave to open as a valid behaviour. It is no less secure; it just means that the Auth Cert will not be useful inside the enclave (unless the user is able to do a similar switch of cards inside).

> After comment from NSA, we note that sufficiently frequent polling of the state of the card reader will ensure that no card swap will have occurred. We therefore conclude that implementing the scenario as stated should not be a problem.

5.2.2 User is denied prohibited initial access to Enclave

Description

A User who should *not* be allowed access to the enclave is prohibited access, making use of biometric authentication.

Stimulus

User inserts a smartcard into the smartcard reader.

Assumptions

ScProhibitInitial.Ass.ValidStart
The ID Station has valid start-up data.

ScProhibitInitial.Ass.ValidConfig
The ID Station has a valid data configuration.

ScProhibitInitial.Ass.Quiescent
The ID Station is quiescent (no other access attempts, configuration changes or start-up activities are in progress).

ScProhibitInitial.Ass.Secure
The User is outside the enclave; the door is closed and locked.

ScProhibitInitial.Ass.FalseUser
The card inserted by the User has an *invalid* ID Certificate, I&A Certificate, or Privilege Certificate, **or** the card inserted by the User has a fingerprint template that is either invalid or *does not* match the fingerprint of the User's finger.

ScProhibitInitial.Ass.PoorAC
The card inserted by the User does not have a valid, current Authorisation Certificate.

Success End-conditions

ScProhibitInitial.Suc.UserCard
The User has possession of the card he originally inserted.

ScProhibitInitial.Suc.PersistCerts
The card inserted by the User contains the same, unchanged data it had at the beginning of the scenario.

ScProhibitInitial.Suc.UserOut
The User is outside the Enclave.

ScProhibitInitial.Suc.Locked
The Enclave door is closed and locked.

ScProhibitInitial.Suc.Audit
The following events have been recorded in the Audit Log (in any order), and the existing audit records are preserved:

- Insertion of card
- Removal of card
- Reading data from card (possibly multiple failures, but at least one success)
- Reading fingerprint image (only if certificates are valid)
- Writing data to the display.
- Validation of any certificate
- Comparison of fingerprint image and template (only if certificates are valid)

Failure Conditions

ScProhibitInitial.Fail.ReadCard
The card inserted by the User does not allow all its data to be successfully read, possibly due to being incorrectly inserted in the first place; being a faulty card; having the incorrect information on it; or being removed before all the information has been read. The set of data to be read is at least:

- ID Certificate
- I&A Certificate
- Privilege Certificate
- Fingerprint Template (contained in the I&A Certificate)

ScProhibitInitial.Fail.Fingerprint
A fingerprint has not been read, possibly due to no finger being presented to the fingerprint reader within X seconds of the display requesting a fingerprint; or the fingerprint not being successfully read within X seconds of the display requesting a fingerprint. The value X shall be taken from configuration data of the ID Station.

ScProhibitInitial.Fail.Audit
Audit files cannot be successfully written. Result: the Door is locked and the system is shutdown.

ScProhibitInitial.Fail.AuditPreserve
Space for audit files has been exhausted. Result: the oldest audit records are overwritten with the new audit records, and an alarm is raised to the Guard.

Constraints

ScProhibitInitial.Con.NoInterleave
No ID Station restart or Configuration data changes will be allowed during this scenario.

Rationale

Auditing of setting the door to locked has been removed, as it is unnecessary to re-set the door to locked if it is already locked. Nothing is gained by choosing to set the door to locked *now*, if we do not trust that it stays locked when we lock it.

Issue

What is the value of “X” in ScProhibitInitial.Fail.Fingerprint?

5.2.3 User gains allowed repeat access to Enclave

Description

A User who should be allowed access to the enclave is given access, but does not use biometric authentication because an Authentication Certificate is found that is still within its validity period.

Stimulus

User inserts a smartcard into the smartcard reader.

Assumptions

ScGainRepeat.Ass.ValidStart
The ID Station has valid start-up data.

ScGainRepeat.Ass.ValidConfig
The ID Station has a valid data configuration.

ScGainRepeat.Ass.Quiescent
The ID Station is quiescent (no other access attempts, configuration changes or start-up activities are in progress).

ScGainRepeat.Ass.Secure
The User is outside the enclave; the door is closed and locked.

ScGainRepeat.Ass.GoodAC
The card inserted by the User has a current, valid Authorisation Certificate.

Success End-conditions

ScGainRepeat.Suc.UserCard
The User has possession of the card he originally inserted.

ScGainRepeat.Suc.PersistCerts
The card inserted by the User contains the same, unchanged data it had at the beginning of the scenario.

ScGainRepeat.Suc.UserIn
The User is in the Enclave.

ScGainRepeat.Suc.Locked
The Enclave door is closed and locked.

ScGainRepeat.Suc.Audit
The following events have been recorded in the Audit Log (in any order), and the existing audit records are preserved:

- Insertion of card
- Removal of card
- Reading data from card (possibly multiple failures, but at least one success)
- Setting the door to locked.
- Setting the door to unlocked.
- Door opening
- Door closing
- Writing data to the display.
- Validation of any certificate (possibly multiple failures, but at least one success)

Failure Conditions

ScGainRepeat.Fail.ReadCard
The card inserted by the User does not allow all its data to be successfully read, possibly due to being incorrectly inserted in the first place; being a faulty card; having the incorrect information on it; or being removed before all the information has been read. The set of data to be read is at least:

- Authorisation Certificate

ScGainRepeat.Fail.UserSlow
The User is too slow in opening the door, so the door locks with the user still outside the enclave. Or the user opens the door, but chooses not to pass through, closing the door again.

ScGainRepeat.Fail.DoorPropped
Once the door has been opened, it is not allowed to close (it is propped open).

ScGainRepeat.Fail.Audit
Audit files cannot be successfully written. Result: the Door is locked and the system is shutdown.

ScGainRepeat.Fail.AuditPreserve
Space for audit files has been exhausted. Result: the oldest audit records are overwritten with the new audit records, and an alarm is raised to the Guard.

Constraints

ScGainRepeat.Con.NoInterleave
No ID Station restart or Configuration data changes will be allowed during this scenario.

Rationale

ScGainRepeat.Ass.ValidUser
This clause has been removed because, following a review by the NSA and a follow-up email on 28/4/2003 at 19.56, it is clear that only the Authorisation Certificate must be read off the card. As soon as a valid, current Authorisation Certificate is found, no further authentication is needed.

Removal of scenario “User is denied prohibited repeat access to Enclave”, because there are now only three possibilities: no valid authorisation certificate (but should be allowed in based on other certificates); no valid authorisation certificate (and should not be allowed in based on other certificates); or has a valid authorisation certificate (and hence should be admitted immediately).

5.2.4 ID Station is started and enrolled with input from the Enrolment Station

Description

A person powers up the ID Station system, and loads the initialisation data from the Enrolment Station via a floppy disk.

Stimulus

Launching the ID Station application from the Windows interface.

Assumptions

ScStart.Ass.NoData
Enrolment data for the ID Station is unavailable internally to the system.

ScStart.Ass.DataOnFloppy
A floppy disk has been inserted into the drive, and the data on the floppy disk from the Enrolment Station is correct.

ScStart.Ass.Secure
The door is closed and locked.

Success End-conditions

ScStart.Suc.Running
The ID Station is running and ready for use, with the data as supplied from the floppy.

ScStart.Suc.Secure
The door is closed and locked.

ScStart.Suc.Audit
The following events have been recorded in the Audit Log (in any order), and the existing audit records are preserved:

- System start-up
- New enrolment data

Failure Conditions

ScStart.Fail.ReadFloppy
The data from the Enrolment floppy is not successfully read. Result: the Door is locked and the system is shutdown.

ScStart.Fail.Audit
Audit files cannot be successfully written. Result: the Door is locked and the system is shutdown.

ScStart.Fail.AuditPreserve
Space for audit files has been exhausted. Result: the oldest audit records are overwritten with the new audit records, and an alarm is raised to the Guard.

Constraints

ScStart.Con.NoInterleave
No ID Station Configuration data changes or User use will be allowed during this scenario.

Issues

How do we distinguish between authorised and non-authorised people? Do we intend that only authorised people will be able to start up the system? (Email from NSA, 24/2/2003).

Answer: we cannot distinguish between authorised and unauthorised people without enrolment data, because it is the presence of the data that defines “authorised” as “known and accepted by the specified authority, which I define for you by giving you keys to check with”.

5.2.5 ID Station is started already enrolled

Description

A person powers up the ID Station system, and the ID Station becomes available for use, as it has previously been enrolled.

Stimulus

Launching the ID Station application from the Windows interface.

Assumptions

ScRestart.Ass.Data
Enrolment data for the ID Station is available internally to the system.

ScRestart.Ass.Secure
The door is closed and locked.

Success End-conditions

ScRestart.Suc.Running
The ID Station is running and ready for use.

ScRestart.Suc.Secure
The door is closed and locked.

ScRestart.Suc.Audit
The following events have been recorded in the Audit Log (in any order), and the existing audit records are preserved:

- System start-up

Failure Conditions

ScRestart.Fail.Data
The internal enrolment data is not internally consistent. Result: the Door is locked and the system is shutdown.

ScRestart.Fail.Audit
Audit files cannot be successfully written. Result: the Door is locked and the system is shutdown.

ScRestart.Fail.AuditPreserve
Space for audit files has been exhausted. Result: the oldest audit records are overwritten with the new audit records, and an alarm is raised to the Guard.

Constraints

ScRestart.Con.NoInterleave
No ID Station Configuration data changes or User use will be allowed during this scenario.

5.2.6 ID Station is shut down

Description

An authorised person powers down the ID Station system.

Stimulus

Command to shut down is typed into the console.

Assumptions

ScShutdown.Ass.Secure
The door is closed and locked.

ScShutdown.Ass.LoggedOn
A Security Officer is currently logged onto the ID Station.

Success End-conditions

ScShutdown.Suc.Shutdown
The ID Station is no longer running and responds to no inputs.

ScShutdown.Suc.Secure
The door is closed and locked.

ScShutdown.Suc.Audit
The following events have been recorded in the Audit Log (in any order), and the existing audit records are preserved:

- Invocation of command to shutdown
- System shutdown

Failure Conditions

ScShutdown.Fail.Audit
Audit files cannot be successfully written. Result: the Door is locked and the system is shutdown.

ScShutdown.Fail.AuditPreserve
Space for audit files has been exhausted. Result: the oldest audit records are overwritten with the new audit records, and an alarm is raised to the Guard.

Constraints

ScShutdown.Con.NoInterleave
No ID Station Configuration data changes or User use will be allowed during this scenario.

Issues

How do we distinguish between authorised and non-authorised people? We intend that only authorised people will be able to shut down the system? (Email from NSA, 24/2/2003).

Answer: we now assume that a security officer is logged on, and so we have been able to block this operation to all other people.

5.2.7 Security Officer updates the configuration of the ID Station

Description

A Security Officer updates the ID Station configuration data with a completely new set of data, from a floppy.

Stimulus

Command to re-configure is typed into the console.

Assumptions

ScConfig.Ass.Secure
The door is closed and locked.

ScConfig.Ass.LoggedOn
A Security Officer is currently logged onto the ID Station.

Success End-conditions

ScConfig.Suc.Config
The ID Station is available for use with its configuration identical to that specified on the floppy.

ScConfig.Suc.Secure
The door is closed and locked.

ScConfig.Suc.Audit
The following events have been recorded in the Audit Log (in any order), and the existing audit records are preserved:

- invocation of command to modify configuration
- Modification of ID Station configuration data values

Failure Conditions

ScConfig.Fail.Read
The configuration data cannot be successfully read from the floppy.

ScConfig.Fail.Audit
Audit files cannot be successfully written. Result: the Door is locked and the system is shutdown.

ScConfig.Fail.AuditPreserve
Space for audit files has been exhausted. Result: the oldest audit records are overwritten with the new audit records, and an alarm is raised to the Guard.

Constraints

ScConfig.Con.NoInterleave
No ID Station shutdown or User use will be allowed during this scenario.

5.2.8 Audit log is archived

Description

An Auditor archives the audit log off the system onto a floppy disk, clearing the audit log on the ID Station.

Stimulus

Command to archive the log is typed into the console.

Assumptions

ScAudit.Ass.Secure
The door is closed and locked.

ScAudit.Ass.LoggedOn
An authorised Auditor is logged on.

Success End-conditions

ScAudit.Suc.Clear
The audit log on the ID Station contains only the audit record relating to this archive action, which are:

- invocation of command to archive audit log
- clearing audit log

- writing audit log to floppy
- read-back confirmation of write to floppy

ScAudit.Suc.Written
The audit log on the ID Station at the beginning of this scenario is on the floppy disk.

Failure Conditions

ScAudit.Fail.Write
The audit log cannot be successfully written to the floppy. Result: failure of the audit archive action is written to the audit log (either preserving the existing audit log if possible, or if not possible, by overwriting the oldest audit records), and alarm is raised to the Auditor.

ScAudit.Fail.Audit
Audit files cannot be successfully written. Result: alarm is raised to the Auditor.

Constraints

ScAudit.Con.NoInterleave
No ID Station shutdown or User use will be allowed during this scenario.

5.2.9 Guard manually unlocks the door

Description

A Guard overrides the latching and requests the door to be unlocked manually to allow the entry of a Person.

Stimulus

Command to unlock the door is typed into the console.

Assumptions

ScUnlock.Ass.LoggedOn
The Guard is logged on.

ScUnlock.Ass.Quiescent
The ID Station is quiescent (no other access attempts, configuration changes or start-up activities are in progress).

ScUnlock.Ass.Secure
The User is outside the enclave; the door is closed and locked.

Success End-conditions

ScUnlock.Suc.UserIn
The User is in the Enclave.

ScUnlock.Suc.Locked
The Enclave door is closed and locked.

ScUnlock.Suc.Audit
The following events have been recorded in the Audit Log (in any order), and the existing audit records are preserved:

- invocation of command to manually unlock door
- Setting the door to locked.
- Setting the door to unlocked.
- Recognising the door has opened.
- Recognising the door has closed.
- Writing data to the display.

Failure Conditions

ScUnlock.Fail.UserSlow
The User is too slow in opening the door, so the door locks with the user still outside the enclave. Or the user opens the door, but chooses not to pass through, closing the door again.

ScUnlock.Fail.DoorPropped
Once the door has been opened, it is not allowed to close (it is propped open).

ScUnlock.Fail.Audit
Audit files cannot be successfully written. Result: the Door is locked and the system is shutdown.

ScUnlock.Fail.AuditPreserve
Space for audit files has been exhausted. Result: the oldest audit records are overwritten with the new audit records, and an alarm is raised to the Guard.

Constraints

ScUnlock.Con.NoInterleave
No ID Station shutdown or User use will be allowed during this scenario.

5.2.10 Administrator logs on

Description

An Administrator logs onto the ID Station by inserting their Token in the Admin Token Reader.

Stimulus

A Token is inserted in the Admin Token Reader.

Assumptions

ScLogOn.Ass.Quiescent
The ID Station is quiescent (no other access attempts, configuration changes or start-up activities are in progress).

ScLogOn.Ass.Secure
The door is closed and locked.

ScLogOn.Ass.ValidAdmin
The card inserted by the Administrator has a valid Authorisation Certificate.

Success End-conditions

ScLogOn.Suc.LogOn
The ID Station is available for use by the Administrator, in that it will respond to the commands allowed to that Administrator as defined by the privileges in the Authorisation Certificate read from the Token and the Configuration data held on the ID Station.

ScLogOn.Suc.Secure
The door is closed and locked.

ScLogOn.Suc.Audit
The following events have been recorded in the Audit Log (in any order), and the existing audit records are preserved:

- Log-on by Administrator
- Insertion of card
- Reading data from card (possibly multiple failures, but at least one success)
- Validation of any certificate (possibly multiple failures, but at least one success)

Failure Conditions

ScLogOn.Fail.ReadCard
The card inserted by the Administrator does not allow all its necessary data to be successfully read, possibly due to being incorrectly inserted in the first place; being a faulty card; having the incorrect information on it; or being removed before all the information has been read. The set of data to be read is at least:

- Authorisation Certificate

ScLogOn.Fail.Audit
Audit files cannot be successfully written. Result: the Door is locked and the system is shutdown.

ScLogOn.Fail.AuditPreserve
Space for audit files has been exhausted. Result: the oldest audit records are overwritten with the new audit records, and an alarm is raised to the Guard.

Constraints

ScLogOn.Con.NoInterleave
No ID Station shutdown or User use will be allowed during this scenario.

Rationale

ScLogOn.Ass.ValidAdmin
Only the Authorisation Certificate is checked, because we assume that the purpose of the Authorisation Certificates is to control access to the workstations within the enclave, and for these purposes the ID Station acts as a workstation. The ID, I&A and Privilege Certificates will have been used to gain entry to the enclave.

5.2.11 Administrator logs off

Description

An Administrator logs off the ID Station.

Stimulus

The Token is removed from the Admin Token Reader.

Assumptions

ScLogOff.Ass.LoggedOn
An Administrator is logged on (which implies an Admin Token is in the Reader).

ScLogOff.Ass.Secure
The door is closed and locked.

Success End-conditions

ScLogOff.Suc.LoggedOff
The ID Station is unavailable for use by anyone at the console; it will respond to no commands typed in at the console.

ScLogOff.Suc.Secure
The door is closed and locked.

ScLogOff.Suc.Audit
The following events have been recorded in the Audit Log (in any order), and the existing audit records are preserved:

- Log-off by Administrator

Failure Conditions

ScLogOff.Fail.Audit
Audit files cannot be successfully written. Result: the Door is locked and the system is shutdown.

ScLogOff.Fail.AuditPreserve
Space for audit files has been exhausted. Result: the oldest audit records are overwritten with the new audit records, and an alarm is raised to the Guard.

Constraints

None.

6 Design Constraints

The system will be developed and run on a workstation running NT 4.0.

7 System Specifications

The interfaces given in section 3.3 are defined fully in reference [1]. Some additional notes are given below.

7.1 Core Functions

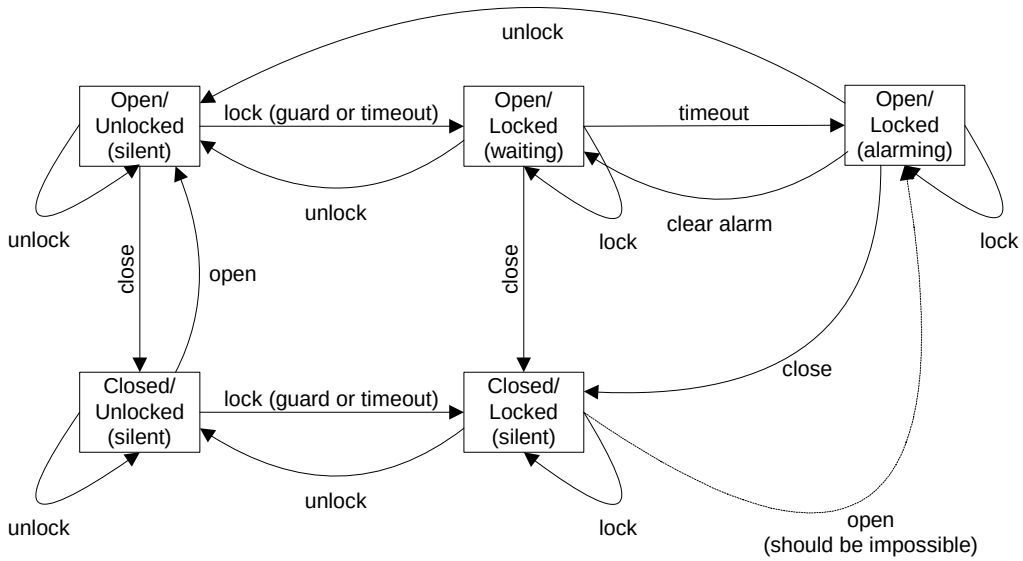
There is no external, callable interface to the core functions. The core is active, and makes calls out to other modules.

7.2 Doors and Alarms

The door behaves as described in the Domain Knowledge section. Within the ID Station system, though, its behaviour can be expanded to include timeout-activated locking and alarms

There is an alarming state: if the door is secure, then the alarm is silent. If the door is potentially insecure (it is open but locked, waiting for a user to pass through before closing the door and becoming secure) then the alarm is silent, but is waiting a timeout period before alarming. If the timeout period passes, the alarm goes off.

As unlocked states are potentially insecure, there is always a time-out period, after which the door will be commanded to lock.



Open/Closed, Locked/Unlocked, and Alarm relationships

Document Control and References

Praxis High Integrity Systems Limited, 20 Manvers Street, Bath BA1 1PX, UK.

Changes history

Issue 0.1 (21/2/2003): First draft. Internally reviewed.

Issue 0.2 (23/4/2003): Final internal review draft.

Issue 1.0 (23/4/2003): For final comment from client.

Issue 1.1 (28/4/2003): Change-bar version for comment from client.

Issue 1.2 (13/5/2003): Final chance for comment, following telephone discussions with NSA.

Issue 1.3 (19/8/2008): Updated for public release.

Changes forecast

None.

Document references

1 TIS Interface Specification, S.P1229.41.3, David Painter, version 0.2, 17 April 2003.