Noggin OCA Incident Manager

Security Target

EAL2 augmented with ALC_FLR.1

Version 1.1

June 2010

Document History

Version	Date	Author	Description
1.1	30 June 2010	Owen Prime	Unclassified release

Table of Contents

List of Tables

List of Figures

1 Document introduction

1 This section provides preliminary information and various documenting conventions which do not formally constitute elements of a Security Target but which are used to present the Security Target to the reader, as well as other information which aims at assisting the reader in understanding the ST and the TOE it describes.

1.1 Document conventions

2 Part 2 of the Common Criteria defines an approved set of operations that may be applied to security functional requirements. Following are the approved operations and the document conventions that are used within this ST to depict their application:

- **Assignment.** The assignment operation provides the ability to specify an identified parameter within a requirement. Assignments are depicted using bolded text and are surrounded by square brackets as follows [**assignment**].
- **Selection.** The selection operation allows the specification of one or more items from a list. Selections are depicted using bold italics text and are surrounded by square brackets as follows [***selection***].
- **Refinement.** The refinement operation allows the addition of extra detail to a requirement. Refinements are indicated using bolded text, for **additions**, and strike-through, for ~~deletions~~.
- **Iteration.** The iteration operation allows a component to be used more than once with varying operations. Iterations are depicted by placing a letter at the end of the component identifier as follows FDP_IFF.1a and FDP_IFF.1b.

3 Additionally, the following conventions may be occasionally used:

- **Application note** is an informal explanation by the author of the ST to highlight and explain an unusual or otherwise exceptional wording either in the requirements for an artefact of the ST or in the statement of a specific artefact in the ST.

1.2 Terminology

4 Table 1 identifies and defines the essential terms and abbreviations used in this document.

Table 1 – Terminology

Term	Description
OCA Connect	Contains identities of clients allowed to use the TOE for relaying communication between each other.
2D-MLS	Two-dimensional Multilevel Security model used to implement mandatory access control features in the TOE.
GUI	Graphical User Interface.
ISM	Australian Government Information Security Manual.
OWASP	Open Web Application Security Project (http://www.owasp.org).

RBAC	Role-Based Access Control. A means of simplifying the management of access rights by defining access rights for roles instead of individual users and then assigning each individual user to a specific role which defines his access rights.
MAC	Mandatory Access Control. A method of enforcing multilevel secure access restrictions.

1.3 References

[1] Common Criteria for Information Technology Security Evaluation Part 1: Introduction and General Model, version 3.1 Revision 3, July 2009, CCMB-2009-07-001.

[2] Common Criteria for Information Technology Security Evaluation Part 2: Security functional components, version 3.1 Revision 3, July 2009, CCMB-2009-07-002.

[3] Common Criteria for Information Technology Security Evaluation Part 3: Security assurance components, version 3.1 Revision 3, July 2009, CCMB-2009-07-003.

1.4 Document organisation

5 This document is organized into the following sections:

- Section 1 provides introductory and preliminary explanations and document conventions to assist readers in understanding this ST.
- The assurance families required for fulfilling assurance class ASE (ST Evaluation) at EAL2, excluding the rationales, are covered as follows:
  - i) ASE_CCL.1 (Conformance claims) in Section 3.
  - ii) ASE_ECD.1 (Extended components definition) In Section 6.
  - iii) ASE_INT.1 (ST introduction) in Section 2.
  - iv) ASE_OBJ.2 (Security objectives) in Section 5.
  - v) ASE_REQ.2 (Derived security requirements) in Section 7.
  - vi) ASE_SPD.1 (Security problem definition) in Section 4.
  - vii) ASE_TSS.1 (TOE summary specification) in Section 8.
- The rationales as all presented centrally in Section 9.

2 ST Introduction

6 This section identifies the ST and describes the TOE in a narrative way.

2.1 ST and TOE reference

7 The reference identifying this ST and the TOE it relates to is given in Table 2.

Table 2 – ST and TOE identification information

ST Title	Noggin OCA Incident Manager Security Target
ST Version	1.1, 30-June-2010
TOE Software	Noggin OCA Incident Manager v.1.1.0.0
Assurance Level	EAL2 augmented with ALC_FLR.1
CC Identification	Common Criteria for Information Technology (IT) Security Evaluation, Version 3.1 Revision 3 as defined in [1], [2] and [3].

2.2 TOE overview

2.2.1 Usage and major security features of the TOE

8 The Noggin Organise Communicate Act (OCA) is a database backed web-based application that coordinates communication between clients associated to it and provides a central repository of audit data.

9 In an emergency scenario a number clients, owned and operated by a number of different organizations need to communicate in a reliable manner. Establishing communication directly between the organizations would be inefficient and the complexity of coordination would significantly complicate the management of security within the systems in which the clients reside. The TOE provides a central point of communication and a repository of contacts to simplify the interconnection of systems. It allows the owners of the clients and the systems they reside in to avoid separate logistic and security arrangements for communication between different parties while maintaining the control of the visibility to others and availability to communication of the clients representing their systems.

10 The OCA is built upon the Linux, Apache, MySQL, PHP (LAMP) stack and developed in the PHP programming language for server-side execution. A combination of XHTML, Javascript, CSS, and Flash is used for client-side presentation and execution. SQL is used for database querying and manipulation and PL/SQL is used for database triggers, functions, and procedures. There are also shell script tools for installation and configuration management.

11 The essential security features of the TOE include protection and filtering of communication between clients, provision of a central repository of audit records, and ensuring that the system configuration remains authentic so that only authorized alterations are allowed.

12 A variety of third party products and services are used by the OCA.

13 The TOE is the Basis Application Framework expanded with the Noggin Communications Gateway and consists of the following:

- Basis application framework director – All incoming web requests are received through the framework director which manages user identification and access control, controls the cache, and oversees the application environment.

Noggin OCA Incident Manager

- OCA application – An object-orientated approach for data definition, operation execution, and GUI abstraction layer.
- Basis application framework – A visual layer invoked by the application to render the GUI for the users theme, browser, and device ensuring syntactical correctness and consistency.
- The Noggin Communications Gateway -- a means for clients to communicate in a secure manner using the OCA as an intermediator.

14 Table 3 highlights the security functions and features that the TOE implements.

Table 3 – TOE security functions and features

Security function	TOE security feature
Secure interoperation	The TOE intermediates all communication between interconnected clients and only allows traffic that is acceptable according to the traffic filtering and information flow control rules established by the administrators.
	The TOE intermediates all traffic between clients associated to it and verifies the authenticity of each message. The messages may be encrypted by the underlying operating system. If a message is deemed authentic the TOE adds further authentication data to it and relays the encrypted message to the recipient.
	The TOE provides a directory of participating systems and agencies that own those systems, called OCA Connect. Each agency chooses the characteristics they wish to make accessible and the criteria other systems must fulfill to be allowed to initiate communication.
Authentication	Each user must enter a valid user name and password to identify and authenticate. The OCA also supports SMS authentication which is not part of the TOE.
	Each user name is associated to a role. The roles are defined by administrators and mandate the security profile and access rights of any user entering that role.
Access control	OCA implements 2D-MLS, a two-dimensional Multi-level security model (MLS) which extends a regular MLS to accommodate a non-linear structure to restrict access within single levels for enforcing need to know.
	Levels in 2D-MLS are defined separately for read, and write access. They constitute a partial order so that those subjects with high access level effectively have access to the lower level objects whereas the subjects with a low access level do not have access to the objects with a higher level access level. Parallel levels can be introduced using the second dimension of the 2D-MLS model.
	Each application and operation in the OCA can be restricted to a minimum access level. Only users with that access level or above can access the application/operation.

Security function	TOE security feature
	Each object in the OCA has a READ and a WRITE access level associated with it. These access levels can be set by users when editing the object. Objects may have additional access level options such as SEND on contacts to control which contacts users of an access level may communicate with.
	Only users with access to the READ access level can view and see the existence of the object. Only users with access to the WRITE access level can change the object including any assigned access levels.
	If a user can READ but not WRITE the object, the WRITE access level is shown as NOT PERMITTED. This also holds for any other access level settings such as SEND.
	Assuming the user has WRITE access to the object, they may only assign access levels up to and including their own access level.
	Inbound interoperability messages are processed at the access level of the user doing the processing. In the case of auto-processing via a rule, the access level of the user who created the rule applies.
Security event monitoring	The OCA collects audit data from security relevant events and provides authorised users and administrators with the capability to review the audit data to monitor the usage of the product and detect potential security flaws.

2.2.2 TOE Type

15 The TOE is a software product for facilitating secure interoperation and communication between clients, not readily classifiable to any of the Common Criteria categories. The TOE includes the following:

- The relevant OCA software developed in the PHP programming language for server-side execution.
- The Communications Gateway used for all inbound and outbound communications. The gateway is controlled via a XML-SOAP API over an SSL connection provided by the environment.

2.2.3 Hardware, software and firmware required by the TOE

16 The OCA is used by registered clients to exchange information in a secure manner. Clients are basically web browsers running on hosts separate from the host that houses the TOE. Typically, clients are connected to the TOE over the Internet. Clients do not constitute parts of the TOE.

17 The TOE is a software product. As such, none of the required server hardware or interconnectivity infrastructure constitute parts of the TOE. Any generic X86 architecture hardware can be used to host the TOE. At a minimum, the hardware hosting the TOE must have at least a single quad-core 2.66 GHz CPU, 4Gb RAM and 50 GB storage.

18 The supporting software such as the operating system (Linux) and the PHP engine is required for successful execution and operation of the TOE but do not constitute parts of the TOE.

19 To provide high levels of security, encryption is implemented using the OpenSSL S/MIME toolset with a CA (Certificate Authority) which are both provided by the environment.

20 The OCA is implemented following a pack architecture that allows the system to be extended for various verticals. It also contains a plug-in architecture to add and modify some elements. This approach has been taken so that the core OCA software development is not branched into multiple products. The packs and plug-ins are kept separate from the core components so that they can be easily identified and documented.

21 Currently the main pack is the Incident Manager which adds new applications, document types, and data objects for events, assets, requests, logs, reports, and links. Packs created for specific customers are considered too unique to be included in the base product.

22 The OCA software is supported by a variety of third party products and services. None of them constitute parts of the TOE but are required for successful operation of the TOE.

2.3 TOE Description

2.3.1 Physical scope of the TOE

23 The TOE consists of the components:

- **Basis application framework director**. The Basis application framework director receives all the incoming web requests from the outside entities, i.e. as forwarded from the client's web browser. The Basis application director manages user identification and access control, controls the cache, and controls the application environment.
- **OCA application**. The actual OCA application is an object-orientated software layer for data definition, operation and execution. It also provides a GUI abstraction layer.
- **Basis application framework**. The Basis application framework is a visual layer invoked by the OCA application to render the GUI for the users theme, browser, and device ensuring syntactical correctness and consistency.
- **The Noggin Communications Gateway**. OCA Interoperability provided by the Noggin Communications Gateway is a secure online system for sharing incident-related information between agencies in real-time, regardless of whether they use Noggin's OCA Incident Manager software.

2.3.2 Logical scope of the TOE

24 The logical scope of the TOE concerns with the protection of communication and interoperation of clients associated to the TOE, and with the controlling of access to the TOE to ensure that only legitimate human users gain access to protected files and functions.

25 Protection of communication and interoperation of clients covers ensuring that only legitimate clients may communicate with each other and that the communication is protected so that the clients can depend on the confidentiality of the communicated content and the authenticity of both communicating parties and the communicated content.

26 Controlling access to the TOE covers features which ensure that only authentic and authorized users are granted access to the TOE configuration files and functions.

27 These concerns are addressed by implementing the security functionality and features summarized below.

2.3.2.1 User authentication

28 In order to gain entry to the OCA each user must be authenticated. User authentication is based on a user name and password. Each user must enter a valid user name and password to identify and authenticate prior to any other action.

29 Upon successful authentication, the user is associated to a role type assigned to that user name. Subsequent authorization and access control decisions are made based on the role type entered. TOE administrators can create new role type and assign users to the role types.

30 If the user authentication fails, the security profile of the role associated to the attempted user name is investigated to determine the policy on failed authentications. The security profile contains a definition of the maximum number of consecutive failed authentication attempts. If that number is exceeded, the account associated to the user name is disabled. Depending on the security profile, the account may be reactivated by TOE administrators.

31 Once successfully authenticated, a user is assigned a role which determines their access and security profile throughout the OCA. This security profile defines the following:

- the password policies for the user:
  - i) how frequently the user must change their password,
  - ii) the password strength requirements,
  - iii) allowable failed login attempts before the account is deactivated,
  - iv) allowable password reset mechanisms and mediums, and
  - v) The method of two-factor authentication.
- The access levels the user is assigned.
- Dashboard restrictions.

2.3.2.2 Access levels and access control

32 Access control in the OCA is based on 2D-MLS, a two-dimensional variant of the Multi-level security model (MLS). 2D-MLS adopts the simplicity and transparency of MLS and extends it to accommodate a non-linear structure of access levels.

33 2D-MLS works like MLS in that the levels are partially ordered so that those with the high-level access effectively have access to the lower-level objects whereas those at the lower-level do not have rights to a high-level access. The difference is that levels can be introduced side-by-side to provide exclusion zones.

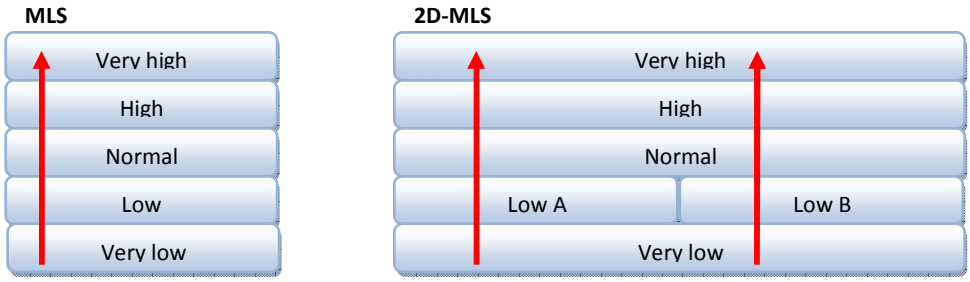

Figure 1 Example of a 2D-MLS scheme

34 Separate access levels are maintained for READ and WRITE accesses. The MLS model illustrated in Figure 1 demonstrates a simple security model for a generic access with a single stream from “Very low” to “Very high” – each level having access to all levels

underneath it. The 2D-MLS model demonstrates a more complex security model with two streams: (1) Very low > Low A > Normal > High > Very high; and (2) Very low > Low B > Normal > High > Very high. In this model:

- Users with “Low A” access also have access to “Very low”, but not “Low B”
- Users with “Low B” access also have access to “Very low”, but not “Low A”
- Users with “Normal” access or above have access to both “Low A” and “Low B”.

Each application and operation in the OCA can be restricted to a minimum access level using the 2D-MLS rules and levels defined by administrators. The access control is implemented at the Basis framework director level which prevents the OCA from event loading the application or executing the operation should the user not have access to it.

35 Some objects may have additional access level options such as SEND on contacts to control which contacts users of an access level may communicate with.

36 In general, the following 2D-MLS rules apply:

- Only users with access to the READ access level can view and see the existence of the object. This includes the object index, related object lists, and the object itself. The following exceptions apply:
  - i) The existence of an object may be visible in audit logs. Therefore, it is recommended that the TOE is configured in a manner that ensures that only explicitly authorized privileged users or administrators have access to the audit logs.
  - ii) The existence of an object may be visible in instantiated workflows.
  - iii) Where a primary key restriction applies (e.g. User names), the denial of a key value may indicate the presence of a record using that value.
- Only users with access to the WRITE access level can change the object including any assigned access levels.
- If a user can READ but not WRITE the object, the WRITE access level is shown as NOT PERMITTED. This is true for any other access level settings such as SEND.
- Assuming the user has WRITE access to the object, they may only assign access levels up to and including their own access level.

37 Object access control is implemented immediately prior to the database interface layer.

38 In the event that a workflow is configured to access or modify object data it acts using the access level defined for that workflow, even if the workflow is triggered by a user with a higher or lower access level.

39 Interoperability subscribers are configured to act at an access level and will only be automatically sent objects up to or at that access level. There are no object access level restrictions when manually pushing an object via interoperability.

40 Inbound interoperability messages are processed at the access level of the user doing the processing. In the case of auto-processing via a rule, the access level of the user who created the rule applies.

2.3.2.3 Audit features

41 The OCA has an audit log to track the actions of users whilst they are logged in. The functions subject to auditing concern with the logging in and out, with sending communication and interoperability messages, and viewing, creation, edition, modification or deletion various objects in the scope of the TOE.

42 The “Viewed” action is only when the object has been opened to see the detail of that object, not if the object appears in a list or index.

43 Each audit log entry stores the identity of the user performing the action, the action that was performed, the date and time of the action, the object which the action was applied to, and either a description of the object that was manipulated or a description of the action that took place.

44 Users with the appropriate access level can view the audit log and filter it by date and time, the user, the action, or the object. If the user has access to the object they will be able to open the object from the log. The audit log is intended to be used in conjunction with the revision history stored with versioned objects to determine exactly what was changed at that time.

45 The clients associated to the TOE do not log in. To handle the generation of audit logs for tracking the client communications, two separate logs are maintained: the audit log which tracks explicit user actions and the interoperability log which tracks what object was sent to which system at what time.

46 The events that trigger an interoperability email being sent are:

- A user explicitly sends the object by pressing a button. In this case the action of them pressing that button is recorded in the audit log as “Sent an interoperability message”. An entry is also added to the interoperability log with the recipient system, the current date and time, the object that was sent, and who sent it.
- A user creates a new object or changes an object that is subscribed to by another system. In this case the action of creating/changing the object is recorded in the audit log as a creation (or an edit) and an entry is added to the interoperability log with the recipient system, the current date and time, and the object that was sent.

2.3.2.4 Secure interoperation

47 Secure interoperation of clients enables secure online sharing of incident-related information between agencies in real-time regardless of whether they use Noggin’s OCA Incident Manager software. By acting as a secure intermediary, the TOE removes the need for agencies to expose their systems to the internet, and ensures that agencies know who they are sharing information with and have complete control over who has access to their data.

48 Secure interoperation provides the following key features:

- A secure directory of participating systems and agencies. Each agency chooses the characteristics of other systems that are allowed to initiate communications.
- A certificate authority (provided by the environment) for participants, to guarantee authenticity of clients and to enable data encryption of payloads.
- A firewall between systems so that individual systems don’t have to be public on the internet to share data.
- A standard data structure for common objects such as incidents, reports, contacts, requests etc.
- Management of inter-system dialogue, offers and acceptances, making it easy to negotiate information sharing arrangements.
- Secure transfer (provided by the environment) f information via point to point encrypted, digitally-signed emails with XML attachments.
- Continuous availability and automatic forwarding of data queued while a participating system is unavailable.
- Retain access to key information from other agencies, even if their systems are offline.

49 For two systems to share data, they must first be introduced and agree on what information they would like to share. OCA Interoperability provides all participating systems with a system directory which will list all other participating systems that have

chosen to be listed. Systems can choose what type of other systems can see them. For example, a system may choose to only be listed for government agencies, or for those in a particular industry. Alternatively, a one-use-only invitation can be created and emailed to someone.

50 Once the two systems can see each other, they can start a dialogue as to what information they want to share. Each system can assign the other system an access level (just like a regular user) and make offers of data. For example, they may offer all contacts in group “XYZ” or all “Media Release” reports. The other system then receives these offers and can choose to “Accept” the offer. The original system is then informed of the acceptance and any future changes to those shared objects will be automatically sent to the receiving system. A full log of what was sent to whom is kept and can be reviewed.

51 Instead of subscribing to explicit data sharing offers, systems can opt to send a contact, document, etc, to another system. To do this, authorized users simply find the data they want to send, select the systems they want to send it to, and press the send button.

52 Received information first appears in the system inbox in the same way as an email or SMS does. When the item is opened users can see what’s inside, who sent it, and when they sent it. If the user decides to import the data they are prompted to answer a series of easy questions like:

- Where should the information be placed? E.g. If it was a contact, what group should we put them in?
- What access level should be assigned to the information?
- Should the incident be merged with another incident with a similar identity?

53 As the fields in each system can be initialized to any specific needs, users are prompted to map the originating systems field to theirs. The answers to all these questions, including the field mappings, are then remembered and used next time. When finished, the user is asked if next time the information should be automatically trusted and processed with no manual intervention.

54 The information is sent by attaching an XML document to a secure email. The sending system (a sending client) encrypts and digitally signs the message. The TOE (Interop system) verifies the signature and checks whether the sending client is allowed to communicate with the intended recipient. If the communication is allowed, the TOE places the encrypted and signed message in the recipient’s inbox. The receiving system (the recipient client) initiates picking up the encrypted information from its inbox on the high-availability system. Once fetched, the digital signature of the message can be verified by the recipient and the payload decrypted.

55 The verification of the signature prior to storing the message in the recipient’s inbox and the checking of whether the sender is allowed the intended communication, together with the creation and storage of related audit records, are included in the TOE. The sender’s and recipient’s actions are done by the corresponding clients and excluded from the scope of the TOE. The encryption and decryption is handled by the underlying operating system.

3 Conformance Claims

56 The following conformance claims are made for the TOE and ST:

- **CCv3.1 Rev.3 conformant.** The ST and the TOE are Common Criteria conformant to Common Criteria version 3.1 Revision 3 defined in [1], [2] and [3].
- **Part 2 conformant.** The ST is Common Criteria Part 2 conformant.
- **Part 3 conformant.** The ST is Common Criteria Part 3 conformant.
- **Package conformant.** The ST is package conformant to the package Evaluation Assurance Level EAL2 augmented with ALC_FLR.1 as defined in [3].
- **Protection Profile conformance.** The ST does claims conformance to the following Protection Profiles: **None.**

4 Security problem definition

57 The TOE is concerned with the protection of assets stated in Table 4.

Table 4 Assets protected by the TOE

Identifier	Asset statement
AST.COMM	The legitimacy of the communication between the TOE and connected client systems. Legitimacy of communication is ensured of client systems only receive and send communication they have explicitly approved, and each communicated message is protected for confidentiality and authenticity.
AST.AUTH	Authenticity of TOE access is ensured when the TOE can reliably differentiate legitimate and illegitimate access requests and only grant access to those parties that are successfully authenticated as parties having legitimate a right to access the TOE.
AST.ACCESS	Legitimacy of TOE access is ensured when each authenticated party can be unambiguously assigned to a role and granted only the minimum set of accesses available to that role as governed by the role based and multilevel access control characteristics of the TOE.
AST.AUDIT	Accuracy of audit data is ensured when each auditable event causes a creation and storage of an audit record, and only authorized administrators of the TOE have a right to alter the audit records as part of legitimate audit trail review.

58 The subjects, some of which constitute threat agents as highlighted in the description of threats, are stated in Table 5.

Table 5 Subjects acting within the TOE

Identifier	Subject statement
S.CLIENT	The client system (technical user) of the TOE, associated to the TOE and using TOE for communication with other client systems.
S.USER	A legitimate end user (human user) of the TOE.
S.ADMIN	A legitimate administrator of the TOE.
S.ATTACKER	A threat agent, an illegitimate party attempting to masquerade as a legitimate S.CLIENT, S.USER or S.ADMIN.

4.1 Threats

59 Threats to security as relevant to the TOE are enumerated in Table 6.

Table 6 – Threats to security

Identifier	Threat statement
T.INFOFLOW	S.ATTACKER succeeds in violating AST.COMM by succeeding in establishing a communication channel between two clients which utilizes the TOE but circumvents or bypasses the controls in place to

Identifier	Threat statement
	ensure only legitimate information flows occur.
T.USER_AUTH	S.ATTACKER succeeds in violating AST.AUTH by successfully guessing a correct password that matches a known user name.
T.ROLE	S.ATTACKER succeeds in violating AST.ACCESS by entering a role S.ADMIN or S.USER without possessing credentials required to enter those roles. Furthermore, S.ATTACKER may succeed in obtaining access rights to the system that should only be available to legitimate user roles (S.ADMIN or S.USER) without entering any of the two roles.
T.PRIVILEGE_ESC	S.USER who does not possess the credentials and authorizations to assume role S.ADMIN succeeds in violating AST.ACCESS or AST.AUDIT by manipulating the role based access control system to acquire credentials required for entering S.ADMIN.
T.ILLEGAL_ACCESS	S.ATTACKER succeeds in violating AST.ACCESS or AST.AUDIT by discovering a means to circumvent of bypass the access control facilities of the TOE to gain illegal access to the controlled data or services.
T.COMM_CONF	S.ATTACKER succeeds in violating AST.COMM by compromising the confidentiality of the messages relayed between client systems by the TOE.
T.AUDIT	S.ATTACKER succeeds in violating AST.AUDIT by manipulating the TOE in a manner that prevents an audit record from being generated for an auditable event.

4.2 Organizational Security Policies

60 Organizational security policies applicable to the TOE are stated in Table 7.

Table 7 – Organizational Security Policies

Identifier	OSP statement
OSP.CRYPTO	The minimum key lengths and cryptographic algorithms as well as assurance requirements for the implementations of them within clients are set and enforced by the administrators prior to accepting registration of clients in the OCA Connect. The clients must use SHA-1, AES with 256-bit keys and RSA with 2048-bit keys to protect the email attachments using S/MIME protocol. The used cryptographic keys must be those generated and distributed to the clients by the TOE. The cryptographic implementations must be approved by the TOE administrators and comply with the Australian Information Security Manual (ISM) and other Australian Government regulations.

4.3 Assumptions

61 Assumptions about the usage and operation of the TOE are enumerated in Table 8.

Table 8 – Assumptions

Identifier	Assumption statement
A.PHYS_SEC	The TOE resides in a physically secure premises governed by appropriate physical, procedural and administrative security arrangements that ensure that only legitimate and authorized administrators can gain physical access to the TOE or the immediate IT support required by the TOE.
A.NTP	The TOE is only operated in association with an underlying operating system that is configured to implement a reliable NTP daemon associated to a trustworthy NTP service so that the resulting time stamps provided for use by the TOE are of sufficient quality to facilitate generation of audit records.
A.CRYPTO	The clients associated to the TOE and using the TOE to relay encrypted email attachments between each other implement good quality cryptographic keys and cryptographic functions so that the confidentiality, authenticity or integrity of the messages cannot be compromised when outside the TOE. Additionally, the environment will provide the necessary cryptographic components to protect the integrity and confidentiality of data.
A.ADMIN	All administrators are assumed to be competent, to follow all guidance, and will maintain the security posture of environment supporting the TOE.

5 Security objectives

63 This section states the exact security objectives for the TOE so that the security problem definition is adequately and completely addressed. The security objectives are stated for the TOE and for the operational environment of the TOE.

5.1 Security objectives for the TOE

64 Security objectives for the TOE are stated in Table 9.

Table 9 – Security objectives for the TOE

Identifier	Objective statement
O.INFOFLOW	Information flows utilizing the TOE only occur between clients that have explicitly allowed such information flows to take place.
O.IDENT_AUTH	Each TOE user must be successfully authenticated prior to granting any access and must hold the appropriate privilege to perform TOE functions.
O.ROLE_ASSIGNMENT	Each authentic user is assigned an account type. Based on that account type, the read and write access level are determined for that user. No other rights than those explicitly stated as applicable to that user.
O.MIN_ACCESS	By default, restrictive access privileges are granted to all users and can only be extended by an authorized administrator or user granted with administrative privileges.
O.AUDIT	Each auditable event produces an audit record which is stored within the TOE.

5.2 Security objectives for the environment

5.2.1 Security objectives for the IT environment

65 Security objectives for the IT environment of the TOE are stated in Table 10.

Table 10 – Security objectives for the IT environment

Identifier	Objective statement
OE.TIME	The time stamps produced by the underlying operating system for use by the TOE are accurate and precise to an extent necessary to facilitate their use for time stamping audit records.
OE.CRYPTO	The clients associated to the TOE and using the TOE will relay encrypted email attachments between each other implementing good quality cryptographic keys and cryptographic functions so that the confidentiality, authenticity or integrity of the messages cannot be compromised when outside the TOE. Additionally, the environment will provide the necessary cryptographic components to protect the integrity and confidentiality of data.

5.2.2 Security objectives for the non-IT environment

66 Security objectives for the non-IT environment of the TOE are stated in Table 11.

Noggin OCA Incident Manager

Table 11 – Security objectives for the non-IT environment

Identifier	Objective statement
OE.PHYS_SEC	The TOE resides in physically secure premises protected by personnel, physical and procedural means so that only authentic administrators can obtain physical access to the TOE.
OE.ADMIN	The owners of the TOE will ensure that all administrators are competent, follow all guidance, and maintain the security posture of environment supporting the TOE.

6 Extended component definition

68 There are no extended components applicable to the TOE; hence none of the requirements for the Extended Components Definition (ASE_ECD) are applicable to this ST.

7 IT Security Requirements

7.1 Overview

69 This section defines the security requirements satisfied by the TOE. Each requirement has been extracted from version 3.1 of the Common Criteria, part 2 providing functional requirements and part 3 providing assurance requirements.

70 The security functional requirements are expressed using the notation stated in Section 1.1 and summarized in Table 12.

Table 12 Summary of the TOE Security Functional Requirements

Identifier	Title
Security audit	
FAU_GEN.1	Audit data generation
FAU_GEN.2	User identity association
FAU_SAR.1	Audit review
FAU_STG.1	Protected audit trail storage
User data protection	
FDP_IFC.1	Subset information flow control (Client Communication SFP)
FDP_IFF.1	Simple security attributes (Client Communication SFP)
FDP_ACC.1a	Subset access control (FEATURE SFP)
FDP_ACF.1a	Security attribute based access control (FEATURE SFP)
FDP_ACC.1b	Subset access control (OBJECT SFP)
FDP_ACF.1b	Security attribute based access control (OBJECT SFP)
Identification and authentication	
FIA_AFL.1	Authentication failure handling
FIA_ATD.1	User attribute definition
FIA_SOS.1	Verification of secrets
FIA_UAU.2	User authentication before any action
FIA_UID.2	User identification before any action
Security management	
FMT_MSA.1a	Management of security attributes (Client Communication SFP)
FMT_MSA.1b	Management of security attributes (FEATURE SFP)
FMT_MSA.1c	Management of security attributes (OBJECT SFP)
FMT_MSA.3a	Static attribute initialization (Client Communication SFP)
FMT_MSA.3b	Static attribute initialization (FEATURE SFP)
FMT_MSA.3c	Static attribute initialization (OBJECT SFP)

FMT_SAE.1	Time-limited authorisation
FMT_SMF.1	Specification of management functions
FMT_SMR.1	Security roles

7.2 Security functional requirement statements

7.2.1 FAU: Security Audit

7.2.1.1 FAU_GEN.1 Audit data generation

Hierarchical to:	No other components.
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 audit functions,~~ b) All auditable events for the [***not specified***] level of audit; and c) [**the following:** **a. Successful login,** **b. Account suspension,** **c. Logging out (explicitly, not via auto-logged out),** **d. Sending of communication,** **e. Sending of an interoperability message,** **f. Viewing, creation, editing, exporting, or deletion of contacts,** **g. Viewing, creation, editing, or deletion of contact groups,** **h. Viewing, creation, editing, or deletion of documents,** **i. Viewing, creation, editing, or deletion of tasks,** **j. Viewing, creation, editing, or deletion of appointments,** **k. Viewing, creation, editing, or deletion of contacts,** **l. Viewing, creation, editing, or deletion of events (incidents),** **m. Viewing, creation, editing, or deletion of reports,** **n. Viewing, creation, editing, or deletion of requests,** **o. An explicit sending of an object by a user, and** **p. A creation of a new object or changing of an object that is subscribed to by another system**].
FAU_GEN.1.2	The TSF shall record within each audit record at least the following information: a) Date and time of the event, type of event, subject identity (if applicable), 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 PP/ST, [**and the following:** a. **which object the action was performed on if applicable, and** b. **A description of either the object or if appropriate a description of what was done**].
Dependencies:	FPT_STM.1 Reliable time stamps
Notes:	Items (a) – (m) enumerated under FAU_GEN.1.1 cause an entry to be added in the audit log and items (n) – (o) cause an entry to be added to the interoperability log. Due to the nature of the TOE, auditing the start-up and shutdown of the TOE is not possible. The TOE is a web application that has no start up and shut down in the context of this SFR. Audit generation is performed by individual functions and is never started or stopped. Functions cannot be executed without an audit record being generated.

7.2.1.2 FAU_GEN.2 User identity association

Hierarchical to:	No other components.
FAU_GEN.2.1	For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.
Dependencies:	FAU_GEN.1 Audit data generation FIA_UID.1 Timing of identification
Notes:	None.

7.2.1.3 FAU_SAR.1 Audit review

Hierarchical to:	No other components.
FAU_SAR.1.1	The TSF shall provide [**administrators and users acting in roles to whom administrators have granted right to read audit records**] with the capability to read [**all fields**] 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.
Dependencies:	FAU_GEN.1 Audit data generation
Notes:	None.

7.2.1.4 FAU_STG.1 Protected audit trail storage

Hierarchical to:	No other components.
FAU_STG.1.1	The TSF shall protect the stored audit records in the audit trail from unauthorized deletion.

FAU_STG.1.2	The TSF shall be able to [***prevent***] unauthorized modification to the stored audit records in the audit trail.
Dependencies:	FAU_GEN.1 Audit data generation.
Notes:	None.

7.2.2 FDP: User Data Protection

7.2.2.1 FDP_IFC.1 Subset information flow control (Client Communication SFP)

Hierarchical to:	No other components
FDP_IFC.1.1	The TSF shall enforce the [**Client Communication SFP**] on [**a) Subjects:** **i. Sender,** **ii. Recipient** **b) Information:** **i. Message** **c) Operations:** **i. SEND**].
Dependencies:	FDP_IFF.1 Simple security attributes
Notes:	None.

7.2.2.2 FDP_IFF.1 Simple security attributes (Client Communication SFP)

Hierarchical to:	No other components
FDP_IFF.1.1	The TSF shall enforce the [**Client Communication SFP**] based on the following types of subject and information security attributes: [**a) Sender and Recipient:** **i. Account status,** **ii. Registered user certificate,** **iii. Registered user email address,** **iv. List of clients from whom communication is accepted].** **b) Message:** **i. 'From' field** **ii. 'To' field**].
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: [**a) Both the sender and recipient have active gateway accounts with a current security (X.509v3) certificate registered with the OCA Connect directory subsystem;** **b) The message "From" address matches the sending gateway**

	account's primary email address AND matches the email specified in the sender's X.509v3 certificate; **c) The message is "To" a single recipient, specified by an email address that is a recipient's primary email address registered in the OCA Connect directory subsystem;** **d) The message is NOT to a recipient that the sender cannot see in the OCA Connect directory subsystem.** **e) The message is NOT to a recipient that indicates they are not accepting interoperability messages in the OCA directory subsystem.** **f) All security envelopes contained in the message are signed using the sender's certificate and private key.** **g) The sender's certificate is embedded in the signature.** **h) The sender's certificate is valid (Signed by the OCA Connect CA, not revoked and not expired)**].
FDP_IFF.1.3	The TSF shall enforce the [**None**].
FDP_IFF.1.4	The TSF shall provide the following [**None**].
FDP_IFF.1.5	The TSF shall explicitly authorise an information flow based on the following rules: [**None**].
FDP_IFF.1.6	The TSF shall explicitly deny an information flow based on the following rules: [**None**].
Dependencies:	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialization
Notes:	None.

7.2.2.3 FDP_ACC.1a Subset access control (FEATURE SFP)

Hierarchical to:	No other components.
FDP_ACC.1a.1	The TSF shall enforce the [**FEATURE SFP**] on [**a) Subjects:** **i. Users,** **b) Objects:** **i. Feature/Application,** **c) Operations:** **i. ACCESS**].
Dependencies:	FDP_ACF.1 Security attribute based access control
Notes:	None.

7.2.2.4 FDP_ACF.1a Security attribute based access control (FEATURE SFP)

Hierarchical to:	No other components.

FDP_ACF.1a.1	The TSF shall enforce the [**FEATURE SFP**] to objects based on the following: [**a) All subjects:** **i. Account type (inheriting access levels);** **b) All objects:** **i. Access Level**].
FDP_ACF.1a.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**a) A subject is allowed to ACCESS an object if and only if the access level of that object is less than or equal to the access level of that subject**].
FDP_ACF.1a.3	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**None**].
FDP_ACF.1a.4	The TSF shall explicitly deny access of subjects to objects based on the [**None**].
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
Notes:	An equal access level of a subject refers to the exact same access level. For example, LOW A is equal to LOW A and not LOW B is equal to LOW A. This is further demonstrated in Section 2.3.2.2, Access levels and access control.

7.2.2.5 FDP_ACC.1b Subset access control (OBJECT SFP)

Hierarchical to:	No other components.
FDP_ACC.1b.1	The TSF shall enforce the [**OBJECT SFP**] on [**a) Subjects:** **i. User,** **b) Objects:** **i. OCA object,** **c) Operations:** **i. READ;** **ii. WRITE; and** **iii. SEND (dependant on the object)].**
Dependencies:	FDP_ACF.1 Security attribute based access control
Notes:	The SEND operation only applies to specific objects such as contacts.

7.2.2.6 FDP_ACF.1b Security attribute based access control (OBJECT SFP)

Hierarchical to:	No other components.
FDP_ACF.1b.1	The TSF shall enforce the [**OBJECT SFP**] to objects based on the following: [**a) All subjects:** **i. Account type (inheriting access levels);** **b) All objects:** **i. READ access Level,** **ii. WRITE access Level,** **iii. SEND access Level**].
FDP_ACF.1b.2	The TSF shall enforce the following rules to determine if an operation among controlled subjects and controlled objects is allowed: [**a) A subject is allowed to READ an object if the READ access level of that object is less than or equivalent to the access level of that subject;** **b) A subject is allowed to WRITE an object if the WRITE access level of that object is less than or equivalent to the access level of that subject; and** **c) A subject is allowed to SEND an object if the SEND access level of that object is less than or equivalent to the access level of that subject**].
FDP_ACF.1b.3	The TSF shall explicitly authorise access of subjects to objects based on the following additional rules: [**None**].
FDP_ACF.1b.4	The TSF shall explicitly deny access of subjects to objects based on the [**None**].
Dependencies:	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialization
Notes:	Only users with access to the READ access level can view and see the existence of the object. This includes the object index, related object lists, and the object itself. The following exceptions apply: • The existence of an object may be visible in audit logs; • The existence of an object may be visible in instantiated workflows; and • Where a primary key restriction applies (e.g. User names), the denial of a key value may indicate the presence of a record using that value. An equal access level of a subject refers to the exact same access level. For example, LOW A is equal to LOW A and not LOW B is equal to LOW A. This is further demonstrated in Section 2.3.2.2, Access levels and access control.

7.2.3 FIA: Identification and authentication

7.2.3.1 FIA_AFL.1 Authentication failure handling

Hierarchical to:	No other components.
FIA_AFL.1.1	The TSF shall detect when [***an administrator configurable positive integer within*** [**1 through 32768**]] unsuccessful authentication attempts occur related to [**User Authentication**].
FIA_AFL.1.2	When the defined number of unsuccessful authentication attempts has been [***surpassed***], the TSF shall [**deactivate the account**].
Dependencies:	FIA_UAU.1 Timing of authentication
Notes:	Administrators are responsible for defining the security parameters concerning the number of authentication attempts available to those attempting to authenticate as TOE users (end users or administrators). The TOE keeps track of the number of consecutive failed authentication attempts for each account and if the number is exceeded, the account to which the login attempt occurred is locked.

7.2.3.2 FIA_ATD.1 User attribute definition

Hierarchical to:	No components.
FIA_ATD.1.1	The TSF shall maintain the following list of security attributes belonging to individual **TOE** users: [**a) User account type;** **b) Username;** **c) Password;** **d) Access level; and** **e) Account status**].
Dependencies:	No dependencies.
Notes:	The access level (READ and WRITE) specifies the access level required to see and change the user's account details.

7.2.3.3 FIA_SOS.1 Verification of secrets

Hierarchical to:	No other components.
FIA_SOS.1.1	The TSF shall provide a mechanism to verify that secrets meet [**the following quality metrics for user authentication:** **a) must contain a configurable minimum number of characters (as specified by the Enterprise Administrator);** **b) must include at least one alpha, 1 numeric and 1 symbolic characters;** **c) must not contain a repeating predictable sequence; and** **d) password not contained in dictionary (as specified by TOE dictionaries)**].

Dependencies:	No dependencies.
Notes:	While the TOE has the mechanisms to implement the above quality checks for secrets, these settings are configurable by the administrator and may be enforced based on the user account type.

7.2.3.4 FIA_UAU.2 User authentication before any action

Hierarchical to:	FIA_UAU.1 Timing of authentication
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.
Dependencies:	FIA_UID.1 Timing of identification
Notes:	None.

7.2.3.5 FIA_UID.2 User identification before any action

Hierarchical to:	FIA_UID.1 Timing of identification
FIA_UID.2.1	The TSF shall require each user to be successfully identified before allowing any other TSF-mediated actions on behalf of that user.
Dependencies:	No dependencies.

7.2.4 FMT: Security management

7.2.4.1 FMT_MSA.1a Management of security attributes (Client Communication SFP)

Hierarchical to:	No other components.
FMT_MSA.1a.1	The TSF shall enforce the [**Client Communication SFP**] to restrict the ability to [***modify***] the security attributes [**a) Account status, registered user certificate, registered user email address, list of clients from whom communication is accepted; and** **b) Email 'FROM' field; and Email 'TO' field.**] to [**a) Administrator or user who has been granted the feature;** **b) Message creator**].
Dependencies:	[FDP_ACC.1 Subset access control, FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
Notes:	None.

7.2.4.2 FMT_MSA.1b Management of security attributes (FEATURE SFP)

Hierarchical to:	No other components.

FMT_MSA.1b.1	The TSF shall enforce the [**FEATURE SFP**] to restrict the ability to [***modify***] the security attributes [**Account type (inheriting access level), Access Level**] to [**Administrators**].
Dependencies:	[FDP_ACC.1 Subset access control, FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
Notes:	None.

7.2.4.3 FMT_MSA.1c Management of security attributes (OBJECT SFP)

Hierarchical to:	No other components.
FMT_MSA.1c.1	The TSF shall enforce the [**OBJECT SFP**] to restrict the ability to [***modify***] the security attributes [**a) Object access level** **b) Account type (inheriting access level)**] to [**a) Administrator or users with account type containing a access level equal to or higher than the objects WRITE access level;** **b) Administrators**].
Dependencies:	[FDP_ACC.1 Subset access control, FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions
Notes:	None.

7.2.4.4 FMT_MSA.3a Static attribute initialization (Client Communication SFP)

Hierarchical to:	No other components.
FMT_MSA.3a.1	The TSF shall enforce the [**Client Communication SFP**] to provide [***permissive***] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3a.2	The TSF shall allow the [**object creator**] to specify alternative initial values to override the default values when an object or information is created.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
Notes:	

7.2.4.5 FMT_MSA.3b Static attribute initialization (FEATURE SFP)

Hierarchical to:	No other components.
FMT_MSA.3b.1	The TSF shall enforce the [**FEATURE SFP**] to provide [***restrictive***] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3b.2	The TSF shall allow the [**None**] to specify alternative initial values to override the default values when an object or information is created.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
Notes:	None.

7.2.4.6 FMT_MSA.3c Static attribute initialization (OBJECT SFP)

Hierarchical to:	No other components.
FMT_MSA.3c.1	The TSF shall enforce the [**OBJECT SFP**] to provide [***restrictive***] default values for security attributes that are used to enforce the SFP.
FMT_MSA.3c.2	The TSF shall allow the [**None**] to specify alternative initial values to override the default values when an object or information is created.
Dependencies:	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles
Notes:	The READ, WRITE and SEND permissions of an object match the access level of the object creator (user).

7.2.4.7 FMT_SAE.1 Time-limited authorisation

Hierarchical to:	No other components.
FMT_SAE.1.1	The TSF shall restrict the capability to specify an expiration time for [**user passwords**] to [**administrators**].
FMT_SAE.1.2	For each of these security attributes, the TSF shall be able to [**enforce a password change**] after the expiration time for the indicated security attribute has passed.
Dependencies:	FMT_SMR.1 Security roles FPT_STM.1 Reliable time stamps
Notes:	Password expiration settings are linked to the user account type.

7.2.4.8 FMT_SMF.1 Specification of management functions

Hierarchical to:	No other components.
FMT_SMF.1.1	The TSF shall be capable of performing the following security management functions: [**a) Define roles and associate users to roles, - users to user**

	account type; **b) Define user status;** **c) Establish password quality criteria inherited from the user account type;** **d) Review audit trails - logs,** **e) Modify the presence of clients in the OCA connect,** **f) Modify the values in the clients' List of clients from whom communication is accepted from,** **g) Modify the write, read and send access levels of objects].** **h) Access level of features within the OCA;** **i) Modify the READ WRITE and SEND access level of objects;** **j) Modify the ACCESS access level of features;** **k) Defining access levels.**
Dependencies:	No dependencies.
Notes:	None.

7.2.4.9 FMT_SMR.1 Security roles

Hierarchical to:	No other components.
FMT_SMR.1.1	The TSF shall maintain the roles [**a) User (that inherits access levels from account type); and** **b) Administrator**].
FMT_SMR.1.2	The TSF shall be able to associate users with roles.
Dependencies:	FIA_UID.1 Timing of identification
Notes:	None.

7.3 TOE Security Assurance Requirements

71 The assurance package for the evaluation of the TOE is Evaluation Assurance Level 2 (EAL2) augmented with ALC_FLR.1 (Basic flaw remediation).

72 EAL2 assurance requirements provide confidence in the security functionality of the TOE by analysis using a functional and interface specification, guidance documentation and the high-level design of the TOE, to understand the security behaviour.

73 The analysis is supported by independent testing of the TOE security functions, evidence of developer testing based on the functional specification, selective independent confirmation of the developer test results, strength of function analysis, and evidence of a developer search for obvious vulnerabilities.

74 EAL2 also provides assurance through a configuration list for the TOE, and evidence of secure delivery procedures.

75 The developer monitors the operating domain of the TOE regularly and issues upgrades to the TOE if necessary. The methods of issuing and installing the upgrades are not part of the functionality of the TOE but the upgrades are used for correcting known security

vulnerabilities or other flaws that may result in security vulnerabilities. These procedures are subjected to further assurance by the inclusion of ALC_FLR.1.

76 Table 13 below provides a summary of the TOE security assurance requirements for this evaluation. Complete details of all assurance components are located in part 3 of the Common Criteria.

Table 13 – Summary of TOE security assurance requirements

Assurance class	Assurance components
Development (ADV)	ADV_ARC.1 Security architecture description
	ADV_FSP.2 Security-enforcing functional specification
	ADV_TDS.1 Basic design
Guidance Documents (AGD)	AGD_OPE.1 Operational user guidance
	AGD_PRE.1 Preparative procedures
Life-Cycle Support (ALC)	ALC_CMC.2 Use of a CM system
	ALC_CMS.2 Parts of the TOE CM coverage
	ALC_DEL.1 Delivery Procedures
	ALC_FLR.1 Flaw remediation
Security Target Evaluation (ASE)	ASE_CCL.1 Conformance claims
	ASE_ECD.1 Extended components definition
	ASE_INT.1 ST introduction
	ASE_OBJ.2 Security objectives
	ASE_REQ.2 Derived security requirements
	ASE_SPD.1 Security problem definition
	ASE_TSS.1 TOE summary specification
Tests (ATE)	ATE_COV.1 Evidence of coverage
	ATE_FUN.1 Functional testing
	AET_IND.2 Independent testing – sample
Vulnerability Assessments (AVA)	AVA_VAN.2 Vulnerability analysis

8 TOE Summary Specification

8.1 Overview

77 This chapter provides the TOE summary specification, a high-level definition of the security functions claimed to meet the security functional requirements.

8.2 Security functions

78 The TOE security functions include the following:

- **Secure interoperation.** The TOE provides a means for clients to interoperate in a secure manner.
- **User authentication.** The TOE provides a reliable means to authenticate human users and ensure that only authentic users are granted access to the TOE.
- **2D-MLS access control**. The TOE provides a two-dimensional multi-level secure access control facility that extends and adds flexibility to the common multilevel secure (MLS) access control systems.
- **Security audit**. The TOE provides a means to ensure that each auditable event generates an audit record and that audit records can only be removed by legitimate and authorized administrators.

8.2.1 Secure interoperation

79 The Secure interoperation provided by the TOE concerns with ensuring that communication only takes place between clients that have explicitly been allowed to communicate with each other.

80 Secure interoperation also ensures that when the TOE is used for relaying communication between client systems, the relaying only takes place when:

i) Both the sender and recipient have active gateway accounts with a current security (X.509v3) certificate registered with the OCA Connect directory subsystem;

ii) The message “From” address matches the sending gateway account's primary email address AND matches the email specified in the sender's X.509v3 certificate;

iii) The message is “To” a single recipient, specified by an email address that is a recipient’s primary email address registered in the OCA Connect directory subsystem;

iv) The message is NOT to a recipient that the sender cannot see in the OCA Connect directory subsystem;

v) The message is NOT to a recipient that indicates they are not accepting interoperability messages in the OCA directory subsystem;

vi) All security envelopes contained in the message are signed using the sender's certificate and private key;

vii) The sender's certificate is embedded in the signature; and

viii) The sender's certificate is valid (Signed by the OCA Connect CA, not revoked and not expired).

81 These requirements are formally expressed in FDP_IFC.1 and FDP_IFF.1.

82 In order to ensure that each information flow that may occur is explicitly approved, by default a client that is newly introduced to the TOE (i.e. whose identity is added to the

OCA Connect) does not have any authorization to communicate with other clients (FMT_MSA.3a). Once operational, the administrators may modify the values to connect the newly added client to other clients (FMT_MSA.1a). Modification of the security attributes is a well-defined management function of the TOE (FMT_SMF.1) and the TOE is capable of differentiating between administrators and end users and restricting the administrative functions only to administrators (FMT_SMR.1).

8.2.2 User authentication

83 The TOE includes a function for identifying and authenticating human users. No functions are available to the users prior to a successful identification (FIA_UID.2) and authentication (FIA_UAU.2). Upon successful authentication, each user is assigned a role and access level (FIA_ATD.1). The assigned roles and access level are further used by access control functions to determine whether that user is granted access to the services they request. Upon successful authentication, each user is assigned a role (FMT_SMR.1).

84 User identification and authentication is based on a user name and password. The user name identifies the user account and the password is used for authenticating the user attempting to access that user account. The passwords are only accepted if they meet specific quality criteria (FIA_SOS.1) and that passwords have to be changed constantly according to a timeframe specified by the administrator (FMT_SAE.1). Any attempt of a human user to set or change a password fails if the password does not meet those quality criteria.

85 The TOE keeps track of the number of failed consecutive authentication attempts for each user account. If the number exceeds the threshold defined by TOE administrators, that account is disabled (FIA_AFL.1) and can no longer be accessed.

8.2.3 2D-MLS Access Control

86 In addition to the role based access control, the TOE implements a variant of the Multilevel Secure (MLS) access control called two-dimensional MLS (2D-MLS). The 2D-MLS introduces a non-linear structure on the usual partial order of MLS structures as illustrated in Figure 1 (page 11).

87 Each subject is assigned an access level according to their role type that inherits an access level mapped within a 2D MLS structure and each object is assigned an access level. These attributes are formally coded in FDP_ACC.1a/b.

88 Upon each relevant access request, the access levels are examined to determine whether the access request fulfils the stored MLS rules. If it does not, access is denied. This is formally expressed in FDP_ACF.1a/b.

89 The TOE enforces an access policy where each new subject is assigned an administratively defined access level and each new object is assigned the access level of the object creator. This is stated in FMT_MSA.3b/c. Legitimate administrators and end users, which are the only valid roles within the TOE (FMT_SMR.1), are allowed to modify those values as appropriate to reflect the rights of subjects and security levels of objects to ensure that the system remains operational as coded in FMT_MSA.1b, FMT_MSA.1c and FMT_SMF.1.

8.2.4 Security audit

90 The TOE ensures that each auditable event indeed generates an audit record and that for each auditable event; the minimum set of data is stored in the audit records. This covers FAU_GEN.1. The two types of logs are maintained by the TOE: audit logs and interoperability logs. The audit logs are used for storing audit records of the actions TOE administrators take and the interoperability logs are used for storing audit records of client communication. For all auditable events the identity of the user causing that event is stored and the resulting audit record is associated to that user (FAU_GEN.2). It could be possible to allow audit records being generated on actions taken by unidentified users

but the TOE requires each user to be identified prior to allowing any access to the TOE services and, hence, the situation where an unidentified user causes an audit record to be created may not occur in practice.

91 There is no automated audit trail analysis by the TOE, but the TOE administrators may grant rights to reviewing the audit trails to various roles throughout the TOE operational phase. All users entering the TOE in roles to which access to audit records is granted are allowed to review the audit trails as stated in FAU_SAR.1. The TOE ensures that only authorized parties, i.e. those entering the TOE in roles to which the administrators have explicitly granted a right to remove audit records (e.g. as part of audit trail reduction) are granted removal access to the audit records. This ensures fulfillment of FAU_STG.1.

9 Rationale

9.1 Conformance claims rationale

92 The Conformance Claim of this ST does not claim conformance to any Protection Profile. Hence, there are no elements to be covered in the conformance claim rationale.

9.2 Security objectives rationale

9.2.1 Security objectives for the TOE

93 Table 14 provides a mapping of the TOE Security objectives and threats and a justification for the mapping.

Table 14 – Mapping of TOE security objectives to threats

Threats	Objective	Justification
T.INFOFLOW	O.INFOFLOW	O.INFOFLOW concerns with ensuring that only legitimate information flows occur between two clients associated to the TOE. An attacker may attempt to discover a flaw in the implementation of information flow controls of the TOE and discover a means to inject messages that are not controlled by the TOE. Successful enforcement of O.INFOFLOW ensures that this scenario will not occur. Hence, enforcing O.INFOFLOW fully prevents T.INFOFLOW from occurring.
T.USER_AUTH	O.IDENT_AUTH	O.IDENT_AUTH concerns with ensuring that the passwords are of good quality and well managed so that any attempt to guess passwords of known end users fail with an overwhelming probability. Each password must meet the minimum metrics for quality. These metrics include password length, allowed retries, life-time and the number of different character groups that must be present in a password. Successful user authentication is required before any access to the TOE is granted. Together these metrics ensure that the passwords are of good quality and any password guessing attack will result in the disabling of the attacked account. They achieve this by reducing the probability of successful password guessing to a sufficiently low level so that all password guessing attacks will fail with an overwhelming probability. Ensuring that all password guessing attempts fail with an overwhelming probability fully enforces O.IDENT_AUTH. Furthermore, enforcing O.IDENT_AUTH ensures that T.USER_AUTH may not occur in practice.
T.ROLE	O.IDENT_AUTH	T.ROLE concerns with two scenarios: with the ability of an attacker to enter the TOE as a user without possessing the necessary credentials and

Threats	Objective	Justification
	O.ROLE_ASSIGNMENT	with the ability of a threat agent to discover a method of assuming a legitimate role without succeeding on falsifying an authentication sequence. The first concern is addressed by ensuring that user authentication is required upon each attempt to enter the TOE in a user role. This is covered of O.IDENT_AUTH is enforced. The second concern involves ensuring that only legitimate role types are assigned to end users and that the only method of entering a role type is through user authentication. This is addressed by O.ROLE_ASSIGNMENT. If both O.IDENT_AUTH and O.ROLE_ASSIGNMENT are preserved, T.ROLE is fully countered. Furthermore, preventing T.ROLE from occurring enforces O.IDENT_AUTH and O.ROLE_ASSIGNMENT.
T.PRIVILEGE_ESC	O.MIN_ACCESS O.AUDIT O.IDENT_AUTH	T.PRIVILEGE_ESC concerns with the escalation of privileges. This refers to a situation where a human user successfully authenticates to the TOE and enters a role with restricted privileges. In the less privileged role, the user succeeds in manipulating the TOE in a manner that grants the user administrative privileges, i.e. privileges that should not be accessible to the role assigned to that user. This would result either in a general unauthorized access or in an unauthorized modification of audit trails. To ensure that T.PRIVILEGE_ESC does not occur, the TOE ensures that O.MIN_ACCESS is enforced so that each user is only assigned minimum privileges needed for performing tasks available for the role and additional privileges are only granted upon successfully re-authenticating as a human user allowed to enter the TOE in a role with higher privileges. Ensuring that O.MIN_ACCESS is enforced guarantees that T.PRIVILEGE_ESC may not occur in general TOE access. Furthermore, to ensure that audit trails remain protected from unauthorized access which could provide an attacker with a means to hide any malicious acts, the TOE enforces O.IDENT_AUTH. The aspects of O.IDENT_AUTH relevant to T.PRIVILEGE_ESC concern with ensuring that only legitimate administrators do have a right to modify audit trails and any user without sufficient privileges will fail in attempts to do so. If the access rights to audit records are properly configured and the TOE prevents

Threats	Objective	Justification
		illegitimate accesses to protected objects, then T.PRIVILEGE_ESC is sufficiently prevented as relevant to O.IDENT_AUTH. Furthermore, if O.IDENT_AUTH is fully enforced, the privilege escalation in relation to audit records may occur and T.PRIVILEGE_ESC is prevented from occurring.
T.ILLEGAL_ACCESS	O.IDENT_AUTH O.AUDIT	T.ILLEGAL_ACCESS concerns with the ability of attackers to discover ways of accessing the TOE without possessing the necessary credentials. This concerns with two aspects: ensuring that only upon successful authentication is any access granted to the human users and that upon successful authentication, only those accesses that the human user is entitled to are made accessible to the human user. The first aspect is addressed if O.IDENT_AUTH is preserved and O.AUDIT (with respect to access to audit trails) is preserved. Together, they prevent T.ILLEGAL_ACCESS from occurring. Furthermore, preventing T.ILLEGAL_ACCESS from occurring enforces O.IDENT_AUTH and O.AUDIT by ensuring that no situation could occur where an unauthenticated user gains access to the TOE, a general illegal access is granted by the TOE, or an illegal access to audit records is granted by the TOE.
T.AUDIT	O.AUDIT	T.AUDIT concerns with ensuring that each auditable event indeed results in the generation of an audit record. By enforcing O.AUDIT, the TOE ensures that each auditable event is guaranteed to generate an audit record that is usable in reviewing the events that have taken place within the TOE. Furthermore, preventing T.AUDIT from occurring ensures that no situation where an auditable event does not result in a corresponding audit record may not occur. Therefore, preventing T.AUDIT from occurring enforces O.AUDIT from relevant aspects.

9.2.2 Security objectives for the non-IT environment

94 Table 15 provides a mapping for the assumptions, threats and organizational security policies to the security objectives for the non-IT environment of the TOE.

Table 15 Security objectives rationale for the non-IT environment

Assumption/Threat/OSP	Objective	Justification
A.PHYS_SEC	OE.PHYS_SEC	A.PHYS_SEC concerns with the security measures to ensure that the TOE resides in physically secure premises so that only

		authorized administrators have physical access to the TOE. Such measures are a myriad of physical, procedural and administrative security policies, procedures and measures. As such, they all aim at providing non-IT measures required for enforcing OE.PHYS_SEC that is enforced if the physical premises of the TOE are appropriately secure. Ensuring that A.PHYS_SEC is addressed by the non-IT environment of the TOE enforces OE.PHYS_SEC. Furthermore, enforcing OE.PHYS_SEC addresses all aspects of A.PHYS_SEC and fully covers the assumption.
A.ADMIN	OE.ADMIN	OE.ADMIN requires that the Administrators are competent, follow all guidance, and maintain the security posture of environment supporting the TOE. This objective fully addresses the assumption.

9.2.3 Security objectives for the IT environment

95 Table 16 provides a mapping for the assumptions, threats and organizational security policies to the security objectives for the IT environment of the TOE.

Table 16 – Mapping of objectives rationale for the IT environment

Assumption/Threat/OSP	Objective	Justification
T.COMM_CONF	OE.CRYPTO	The confidentiality of the communication between clients, as relayed by the TOE, concerns with the implementation of proper cryptographic primitives within clients. The TOE does not decrypt and re-encrypt the data but only verifies the authenticity of the messages (email attachments) received from sending clients and adds authentication data to them when forwarding them to the recipients. As such, the confidentiality of the communication depends on the ability of the clients to properly encrypt the messages they communicate with each other using the TOE. While the TOE generates asymmetric key pairs and forwards them to the clients (in form of X.509v3 certificates), it does not participate in the encryption of the email attachments. Therefore, it is up to the IT environment of the TOE (i.e. the Clients) to ensure that the confidentiality of the communication is sufficiently protected. This is articulated in OE.CRYPTO. By enforcing OE.CRYPTO, i.e. ensuring that the clients only use good quality cryptographic primitives and their implementations in the TOE-relayed communication, T.COMM_CONF is fully prevented.

		Furthermore, preventing T.COMM_CONF from occurring ensures that all communication is properly encrypted and enforces OE.CRYPTO.
T.AUDIT	OE.TIME	Prevention of T.AUDIT from occurring requires that the audit records resulting from auditable events are of sufficient quality to be useful in administering the TOE. This requires, among other things, that the time stamps used for labelling the audit records are accurate to the level sufficient for analysing the occurrence and ordering of auditable events that have taken place. As the TOE is an application level component, it does not include its own time management routine but relies in the IT environment, namely the underlying operating system, for the provision of time stamps. The requirement for the accuracy of time stamps is expressed in OE.TIME and enforcement of OE.TIME (i.e. ensuring that the time stamps provided by the IT environment of the TOE are of high quality) is essential for full prevention of T.AUDIT from occurring. Furthermore, enforcing OE.TIME (i.e. ensuring that the time stamps are of high quality) guarantees that the audit records are of high quality and can be used to analyse the event history of the TOE in a reliable manner. Therefore, enforcing OE.TIME counters T.AUDIT from relevant parts.
OSP.CRYPTO A.CRYPTO	OE.CRYPTO	In order to preserve OE.CRYPTO, the use of cryptography must be regulated within the clients. As the TOE does not include the clients, the regulation of the cryptographic processing within the clients must be regulated via an organizational security policy. While the TOE facilitates secure communication by generating some cryptographic keys to the clients, it cannot enforce their use by technical means. The OSP.CRYPTO states the minimum cryptographic key and algorithm requirements for the clients as well as the minimum assurance requirements that the implementations of the cryptographic functions must fulfill. As such, it makes a direct contribution towards OE.CRYPTO by contributing to the cryptographic strength of the protection of communication. The TOE relies in the underlying operating system for cryptographic functions. In order to meet the assumption A.CRYPTO the objective OE.CRYPTO will provide the necessary cryptographic components to protect the integrity and confidentiality of data.

		Furthermore, OE.CRYPTO is satisfied when the cryptographic processing is of high quality and sufficiently strong. When OE.CRYPTO is satisfied, OSP.CRYPTO and A.CRYPTO is also properly enforced.
A.NTP	OE.TIME	The TOE relies in the underlying operating system for reliable time stamps that are used for labelling the audit records. The underlying operating system is a general purpose operating system which cannot realistically be assumed to be a trusted IT product but relies in the expertise of administrators for secure configuration and operation. As such, the NTP daemon of the underlying operating system can only be assumed to be trustworthy and associated to a reliable source of time. No explicitly stated security requirements can be imposed on the NTP daemon. As such, OE.TIME is enforced if the assumption A.NTP is fulfilled and the NTP is properly configured and operated. Furthermore, enforcing OE.TIME ensures that the NTP protocol of the underlying operating system is properly configured and associated to a reliable source of time. Hence, OE.TIME fully addresses A.NTP.

9.3 Security requirements rationale

9.3.1 SFR and SAR Dependency rationale

96 This section demonstrates the SFR and SAR dependencies that are supported by the TOE and provides a justification for the exclusion of dependencies that are not supported by the TOE.

9.3.1.1 Supported dependencies

97 Table 17 demonstrates the mutual supportiveness of the SFR's for the TOE by demonstrating how the SFR dependencies are fulfilled by the TOE, and by justifying those dependencies that are not fulfilled.

98 The SARs relevant to the TOE constitute an evaluation assurance level EAL2 as defined in Common Criteria and augmentation ALC_FLR.1. Therefore, as a complete evaluation assurance level, they are a mutually supportive set and require no further justification.

Table 17 – TOE SFR dependency demonstration

SFR	Dependency	Inclusion
FAU_GEN.1	FPT_STM.1 Reliable time stamps	Not included
FAU_GEN.2	FAU_GEN.1 Audit data generation FIA_UID.1 Timing of identification	FAU_GEN.1 FIA_UID.2
FAU_SAR.1	FAU_GEN.1 Audit data generation	FAU_GEN.1

Noggin OCA Incident Manager

SFR	Dependency	Inclusion
FAU_STG.1	FAU_GEN.1 Audit data generation	FAU_GEN.1
FDP_ACC.1a	FDP_ACF.1 Security attribute based access control	FDP_ACF.1a
FDP_ACF.1a	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation	FDP_ACC.1 a FMT_MSA.3b
FDP_ACC.1b	FDP_ACF.1 Security attribute based access control	FDP_ACF.1b
FDP_ACF.1b	FDP_ACC.1 Subset access control FMT_MSA.3 Static attribute initialisation	FDP_ACC.1b FMT_MSA.3c
FDP_IFC.1	FDP_IFF.1 Simple security attributes	FDP_IFF.1
FDP_IFF.1	FDP_IFC.1 Subset information flow control FMT_MSA.3 Static attribute initialisation	FDP_IFC.1 FMT_MSA.3a
FIA_AFL.1	FIA_UAU.1 Timing of authentication	FIA_UAU.2
FIA_ATD.1	No dependencies.	N/A
FIA_SOS.1	No dependencies.	N/A
FIA_UAU.2	FIA_UID.1 Timing of identification	FIA_UID.2
FIA_UID.2	No dependencies.	N/A
FMT_MSA.1a	[FDP_ACC.1 Subset access control, FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FDP_IFC.1 FMT_SMR.1 FMT_SMF.1
FMT_MSA.1b	[FDP_ACC.1 Subset access control, FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FDP_ACC.1a FMT_SMR.1 FMT_SMF.1
FMT_MSA.1c	[FDP_ACC.1 Subset access control, FDP_IFC.1 Subset information flow control] FMT_SMR.1 Security roles FMT_SMF.1 Specification of Management Functions	FDP_ACC.1b FMT_SMR.1 FMT_SMF.1

SFR	Dependency	Inclusion
FMT_MSA.3a	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FMT_MSA.1a FMT_SMR.1
FMT_MSA.3b	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FMT_MSA.1b FMT_SMR.1
FMT_MSA.3c	FMT_MSA.1 Management of security attributes FMT_SMR.1 Security roles	FMT_MSA.1c FMT_SMR.1
FMT_SMR.1	FIA_UID.1 Timing of identification	FIA_UID.2
FMT_SAE.1	FMT_SMR.1 Security roles FPT_STM.1 Reliable time stamps	FMT_SMR.1
FMT_SMF.1	No dependencies.	N/A

9.3.1.2 Unsupported dependencies

99 FAU_GEN.1 (Audit data generation) and FMT_SAE.1 (Time-limited authorisation) contains an unsupported dependency to FPT_STM.1 (Reliable time stamps). The TOE is a collection of application level software modules executing on a general purpose Linux operating system. Instead, the underlying operating system includes a NTP implementation that provides a synchronized time for use by the TOE. Therefore, the dependency is not applicable to the TOE but is addressed by the underlying operating system through assumption A.NTP.

9.3.2 Tracing of SFR to security objectives

100 Table 18 provides the mapping of the TOE SFRs and the security objectives for the TOE.

Table 18 – Mapping TOE SFRs to objectives

Objective	SFRs	Demonstration
O.INFOFLOW	FDP_IFC.1 FDP_IFF.1 FMT_MSA.1a FMT_MSA.3a FMT_SMF.1 FMT_SMR.1	Preserving O.INFOFLOW is concerned with ensuring that only approved communication takes place between clients that use TOE to intermediate the communication. Secure interoperation ensures that when the TOE is used for relaying communication between client systems, the relaying only takes place when a) both the sender and the recipient are known to the TOE and registered in the OCA Connect, b) the recipient has explicitly approved the sender as a legitimate communicating party with itself, and c) the recipient is on-line so that the communication shall not be lost. These requirements are formally expressed in FDP_IFC.1 and FDP_IFF.1. In order to ensure that each information flow that may occur is explicitly approved, by default a client that is newly introduced to the TOE (i.e. whose identity is added to the OCA Connect) does not have any authorization to communicate with other clients (FMT_MSA.3a). Once operational, the administrators

Objective	SFRs	Demonstration
		may modify the values to connect the newly added client to other clients (FMT_MSA.1a). Modification of the security attributes is a well-defined management function of the TOE (FMT_SMF.1) and the TOE is capable of differentiating between administrators and end users and restricting the administrative functions only to administrators (FMT_SMR.1). Together these SFRs address the information flow characteristics of O.INFOFLOW. Combined, the SFRs the information flow characteristics of O.INFOFLOW.
O.IDENT_AUTH	FIA_AFL.1 FIA_ATD.1 FIA_SOS.1 FMT_SAE.1 FIA_UAU.2 FIA_UID.2 FAU_STG.1 FAU_SAR.1 FDP_ACC.1a FDP_ACF.1a FDP_ACC.1b FDP_ACF.1b	O.IDENT_AUTH concerns with ensuring that human users of the TOE are properly authenticated prior to the access to the TOE being granted. No functions are available to the users prior to a successful identification (FIA_UID.2) and authentication (FIA_UAU.2). Upon successful authentication, each user is assigned a role type (which inherits access levels) (FIA_ATD.1). The assigned role type (which inherits access levels) are further used by access control functions to determine whether that user is granted access to the services they request. The TOE implements a well-defined 2D-MLS model for restricting access of subjects to objects. This policy is defined based on well-defined security attributes (FDP_ACC.1a/b) and enforced by well-defined access rules (FDP_ACF.1a/b). Furthermore, the TOE ensures that audit trails and audit records are protected so that only legitimate parties (i.e. those to whom the administrators have explicitly granted a right to modify audit records) are granted a right to modify the audit trails and any unauthorized modification is prevented (FAU_SAR.1 and FAU_STG.1). User identification and authentication is based on a user name and password. The user name identifies the user account and the password is used for authenticating the user attempting to access that user account. The passwords are only accepted if they meet specific quality criteria (FIA_SOS.1) and that passwords have to be changed constantly according to a timeframe specified by the administrator (FMT_SAE.1). Any attempt of a human user to set or change a password fails if the password does not meet those quality criteria. The TOE keeps track of the number of failed consecutive authentication attempts for each user account. If the number exceeds the threshold defined by TOE administrators, that account is disabled (FIA_AFL.1) and can no longer be accessed. Together these SFRs fully address user authentication and authorisation and enforce O.IDENT_AUTH.
O.ROLE_ASSIGNMENT	FIA_ATD.1	O.ROLE_ASSIGNMENT concerns with ensuring that each user is assigned a role and each assigned role is

Objective	SFRs	Demonstration
	FMT_SMR.1	well defined. FIA_ATD.1 concerns with ensuring that each user is assigned a role upon successful authentication. This ensures that no users exist within the TOE that have not been assigned a role. FMT_SMR.1 concerns with the specific roles defined for the TOE: End user and Administrator. Each role assigned to an authentic end user is one of these roles. Together, addressing these SFRs enforces O.ROLE_ASSIGNMENT.
O.MIN_ACCESS	FMT_MSA.1b FMT_MSA.1c FMT_MSA.3b FMT_MSA.3b FMT_SMF.1	O.MIN_ACCESS concerns with ensuring that by default, only minimum access to the TOE is granted to new users and that only upon authorized decisions can the default access rights be altered. This concerns with ensuring that the newly created subjects and objects are assigned the minimum privileges (FMT_MSA.3b/c). Once created, the privileges of subjects can only be modified by the administrators (FMT_MSA.1b) and the privileges of objects by the creator of that object (FMT_MSA.1c). Privilege modifications can only occur through well-defined management functions (FMT_SMF.1), no other accesses to the privilege modifications exist. Jointly, these SFRs fully enforce O.MIN_ACCESS.
O.AUDIT	FAU_GEN.1 FAU_GEN.2	O.AUDIT concerns with ensuring that each auditable event indeed produces an audit record within the TOE. This directly enforces FAU_GEN.1 and FAU_GEN.2 by ensuring that each audit record is generated and contains the necessary information to make audit trail analysis possible.

9.3.3 SAR justification

101 The set of SARs selected for the TOE constitute the entire evaluation assurance level EAL2 augmented with ALC_FLR.1 for flaw remediation.

102 Excluding the augmentation, as a basic EAL2 package, the set of SARs is an internally consistent and mutually supportive set of SARs.

103 The TOE is used in a potentially untrusted network environment but in a physically controlled environment. The relevant attack scenarios are logical attacks occurring through the external interfaces of the TOE by malicious software residing in remote hosts and accessing the TOE through the network interface.

104 Attack scenarios concerning internal interfaces are not accessible as access to those interfaces would require physical probing of the TOE. Therefore, attack scenarios concerned with physically probing the TOE with expert skill and resources are not relevant.

105 Consequently, it is sufficient for the TOE to be engineered to demonstrate sufficient assurance against logical attacks by malicious software through externally visible interfaces as demonstrated by EAL2.

Noggin OCA Incident Manager

106 The TOE implements additional measures to allow TOE developer to issue software upgrades to the TOE once issued to the end user. While the TOE implements functional measures to ensure the authenticity of the upgrades, there is also a significant assurance component related to the trustworthiness of the upgrades. The developer must demonstrate a comprehensive set of measures followed to ensure that only legitimate and authentic firmware upgrades are issued for the TOE.

107 The TOE described in this ST, however, does implement such ability and for comprehensive assurance, the basic EAL2 package is augmented with ALC_FLR.1 to ensure that the upgrade procedures are sufficiently trustworthy. As the EAL2 selected for the TOE only provides a baseline of assurance, the developers determine that assurance component ALC_FLR.1 is sufficient for consistency.