CBSA Data-Centric Security NIEM System Architecture Document & Prototype Report

Alan Magar,Alan Clason, Will Coxon, Glen Henderson
Yun Liu, James McAllister, Brent Nordin, Dan Seguin, Prateek Srivastava
Bell/Cord3 Team

Prepared By:
Bell Canada
160 Elgin Street – 17th Floor
Ottawa, ON, K1S 5N4
and
Cord 3 Innovations
209-900 Morrison Drive
Ottawa, ON, K2H 8K7

PWGSC Contract Number: W87714-08FE01/Bel
Technical Authority: Daniel Charlebois, Defence Scientist

Disclaimer: The scientific or technical validity of this Contract Report is entirely the responsibility of the Contractor and the contents do not necessarily have the approval or endorsement of the Department of National Defence of Canada.

Contract Report
DRDC-RDDC-2017-C120
March 2016

© Her Majesty the Queen in Right of Canada, as represented by the Minister of National Defence, 2016

© Sa Majesté la Reine (en droit du Canada), telle que représentée par le ministre de la Défense nationale, 2016

CBSA Data-Centric Security NIEM System Architecture Document & Prototype Report

prepared for

Defence Research and Development Canada
and
Canadian Border Services Canada

prepared by

Bell Canada
160 Elgin Street
17th Floor
Ottawa, Ontario
K1S 5N4

Cord3 Innovation
206-900 Morrison Drive
Ottawa, Ontario
K2H 8K7

Final
22 March 2016

Confidentiality

This document is UNCLASSIFIED.

Contributors

Bell / Cord3 Team	Role
Alan Clason	Testing
Will Coxon	Project Management
Glen Henderson	Design & Development
Yun Liu	Architecture
Alan Magar	Team Lead
James McAllister	Lab Setup
Brent Nordin	Design & Development
Dan Seguin	Design & Development
Prateek Srivastava	Research & Design

Revision Control

Revision	Date	Modifications
Phase 1 Draft	26 November 2015	Draft Report (Phase 1)
Phase 2 Draft	27 January 2016	Draft Report (Phase 2)
Phase 3 Draft	21 March 2016	Draft Report (Phase 3)
Final	22 March 2016	Final Report

Executive Summary

Introduction

The overarching objective of the Canada Border Services Agency (CBSA) Data-Centric Security National Information Exchange Model (NIEM) Prototype is to demonstrate how data-centric security facilitates the secure exchange of standardized information with other organizations subject to policy.

Prototype Overview

Radiation detection equipment is located at marine ports in order to prevent dangerous goods from entering the country. Any shipping containers with an elevated reading, above normal background levels, generate an alarm and undergo a risk assessment and further radiation examination to determine the cause and extent of the radiation. When shipments with high levels of radiation are detected, CBSA informs Other Government Departments (OGDs), and specifically both the Canadian Nuclear Safety Commission (CNSC) and Health Canada (HC), in order to prevent radioactive goods from entering Canada and causing health/safety problems to the general public.

The prototype is based on the CBSA RADNET operational architecture and scenario. However, rather than attempt to re-create the CBSA environment in its entirety, the intent was to prototype a subset of the CBSA environment, along with the two OGDs. The Data-Centric Security Service (DCSS) information protection architecture was included in the prototype in order to secure, and control access to, sensitive radiation scan data both within CBSA and when exchanged with the OGDs.

Primary Objective: Standardized Exchange with OGDs

As part of its response to an anomalous radiation detection event, CBSA sends radiation scan data to OGDs. Currently, this radiation scan data is exchanged with OGDs in its native format as email attachments. Consequently, the primary objective of the prototype was to standardize the exchange of radiation scan data with OGDs. This objective is consistent with CBSA's requirement to *harmonize and optimize information exchanges between CBSA and its partners through adoption of standards.*[1] NIEM, which provides a comprehensive information exchange framework, was the obvious choice to facilitate the standardized

[1] *Information Interoperability – Architecture Vision* **[Reference 3]**

exchange of radiation scan data. Of particular interest for this prototype was the Chemical Biological Radiological Nuclear (CBRN) domain, which builds on the NIEM core and adds data elements for the CBRN space. Specifically, the CBRN domain includes an N.25 Information Exchange Package Documentation (IEPD) that specifies the exchange of radiological information. Unfortunately, the current iteration of N.25 is based on an older version of NIEM that is not aligned with the American National Standards Institute (ANSI) N42.42 output produced by radiation detection devices. The newer version of N.25, which was supposed to be based on the current version of NIEM, and is aligned with the N42.42 radiation scan data, has been delayed with no new release date in sight. Consequently, it proved impossible to map the output from radiation detection devices directly to the N.25 IEPD. Fortunately, N.25 allows for documents to be bundled in an N.25 message as encoded binary data, thereby providing a more reliable and verifiable transport for data than email attachments. Furthermore, N.25 is designed for the CBRN space and has data elements to contain the information unique to that space. Within the prototype, these data elements were used to convey radiation scan metadata.

In order to enable this standardized exchange of radiation scan data with OGDs it was necessary to create a new component; the NIEM Policy Enforcement Point (PEP). While this component is discussed further in the secondary objective, it is worth mentioning that it was responsible for intercepting email sent between CBSA and OGDs, extracting the radiation scan data, transforming the radiation scan data, and then composing new email messages to be sent to the intended recipients with the transformed radiation scan data.

Secondary Objective: Secure Exchange with OGDs

The secondary objective was to secure this standardized exchange of radiation scan data with OGDs. The meaning of “secure” in this context is multi-faceted. Not only will the standardized radiation scan data have an appropriate security label denoting the sensitivity of its content, but the security label will be cryptographically bound to the data using a digital signature. In addition, the digitally signed data will be encrypted to prevent unauthorized disclosure during transit. Lastly, the information release will be mediated according to a central security policy. Security labelling data conforming to North Atlantic Treaty Organization (NATO) STANAG 4774 Confidentiality Metadata Label Syntax was included within the NIEM N.25 document, whereas S/MIME was used to encrypt the data, and strongly bind the security label to the data, as it transits between security domains. Lastly, the release of all information exiting the CBSA domain was mediated according to policy. Specifically, the NIEM PEP prevents the egress of any data until such time as a determination has been made with respect to its releasability. If the policy dictates that the information is releasable, then the Policy Decision Point (PDP) would instruct the PEP to release the information. However, if the policy determines that the information is not

permitted for release, then the PEP would be instructed to prevent its release and an email would be sent to the user informing them of the policy decision.

Tertiary Objective: Secure Scan Data within CBSA

The third, and final, objective was to secure scan data, including access to applications containing scan data, within the CBSA prototype domain environment. This means that all sensitive data must be cryptographically protected when stored on CBSA servers and when in transit, and access to data must be mediated so that only authorized users with the appropriate clearance and need-to-know are able to access the sensitive data. Specifically, data assets that are protected by the DCSS infrastructure are encrypted using symmetric key encryption and a key escrow system so that the release of assets is only done when the request is in compliance with policy. Consequently, attempts by privileged users or malicious attackers to compromise the data would fail. In addition, a prototype of the Incident Reporting Tool (IRT) was developed that restricts unauthorized users (e.g., students) to data entry and prevents them from accessing potentially sensitive information.

Conclusion & Recommendations

The CBSA Data-Centric Security NIEM Prototype demonstrated how data-centric security can facilitate the secure exchange of standardized information with other organizations subject to policy. Specifically, it achieved the following three project objectives; standardized exchange of radiation scan data with OGDs, securing this exchange, and securing scan data, including access to applications containing scan data, within the CBSA prototype domain environment.

It is important to note that while NIEM provides an information exchange framework with which to facilitate data transformation, it does not provide the complete set of standards required for immediate implementation. Consequently, even with NIEM, data transformation initiatives are not trivial and will necessitate significant involvement from a variety of stakeholders within the organization.

In order to capitalize on the capability demonstrated within this prototype, it is recommended that CBSA pilot this capability within a production environment. Specifically, the intent would be to provide a standardized container with which to transport any type of scan data within the CBSA production environment. The DCSS information protection architecture would be used to secure the container, and the sensitive scan data embedded within, regardless of where it is stored or transits within the production environment.

Table of Contents

List of Tables

List of Figures

1 Introduction

1.1 Background

The Canada Border Services Agency's (CBSA) mandate is to manage the nation's borders at ports of entry by administering and enforcing the domestic laws that govern trade and travel, as well as international agreements and conventions. The work of the Canada Border Services Agency includes identifying and interdicting high-risk individuals and goods, working with law enforcement agencies to maintain border integrity and engaging in enforcement activities, including seizure of goods, arrests, detentions, investigations, hearings and removals.[2]

In recent research efforts, CBSA has conducted a proof-of-concept for its SensorNet information architecture. Specifically, the Secure Access Management for Secure Operational Networks (SAMSON)[3] technology demonstrator was used to show how the protection of SensorNet data could be enhanced within the CBSA network environment through the application of data-centric security principles. This proof-of-concept was originally described in *Data-Centric Security for CBSA Operations – SAMSON Database Protection for the CBSA SensorNet* **[Reference 1]**.

Under the auspices of this research CBSA also conducted an examination of the National Information Exchange Model (NIEM) as a potential standard to enable information exchange. Specifically, the investigation assessed how data-centric security is supported through NIEM and how it could enable inter-domain exchanges using trust infrastructures such as SAMSON. It was the position of that report that a data-centric security approach to information protection both leverages the standardization of information exchange transactions based on NIEM and also enhances the protection of, access to and auditing of information assets that are exchanged via NIEM. This investigation was documented in *Data-Centric Security and Information Sharing via NIEM* **[Reference 2]**.

Since the completion of these two reports, CBSA has produced an *Information Interoperability – Architecture Vision* **[Reference 3]** document that includes the following vision statement - *harmonize and optimize information exchanges between CBSA and its partners through adoption of standards.* Furthermore, it recommends as a strategic approach that CBSA projects adopt NIEM and the Government of Canada (GC) Interoperability Framework standards.

[2] https://www.tc.gc.ca/eng/corporate-services/planning-dpr-2013-14-1188.html
[3] SAMSON is now referred to as Data-Centric Security Service (DCSS). This term will be used throughout the report.

1.2 Purpose

This report will document the efforts to prototype a NIEM exchange based on a CBSA radiation scanning scenario. The prototype will leverage the Data-centric Security Service (DCSS) information protection architecture to secure, and control access to, sensitive radiation scan data both within CBSA and when exchanged with Other Government Departments (OGDs). The prototype has been divided into the following three phases:

- Phase 1 – NIEM Exchange;
- Phase 2 – Secure NIEM Exchange; and
- Phase 3 – Policy-Mediated NIEM Exchange.

1.3 Document Structure

The remainder of this report consists of the following sections:

- Section 2.0 – Operational Architecture & Scenario: describes the CBSA operational architecture for the radiation scanning scenario;
- Section 3.0 – High-Level Architecture: describes the high-level architecture for each phase of the prototype development;
- Section 4.0 – System Architecture: details the prototype system architecture;
- Section 5.0 – Phase 1: NIEM Exchange – provides a detailed overview of Phase 1 of the prototype;
- Section 6.0 – Phase 2: Secure NIEM Exchange – provides a detailed overview of Phase 2 of the prototype;
- Section 7.0 – Phase 3: Policy Mediated NIEM Exchange – provides a detailed overview of Phase 3 of the prototype;
- Section 8.0 – Conclusions & Recommendations: summarizes the conclusions and recommendations derived from the development of this report;
- Section 9.0 – References: lists the references cited in this report;
- Annex A – RADNET Response Chart: outlines the steps involved in responding to a radiation detection event;
- Annex B – Phase 1 Test Plan, Procedures & Results: details the test plan, procedures and results for Phase 1;
- Annex C – Phase 2 Test Plan, Procedures & Results: details the test plan, procedures and results for Phase 2;
- Annex D – Phase 3 Test Plan, Procedures & Results: details the test plan, procedures and results for Phase 3;
- Annex E – XSLT Primer: provides an overview of eXtensible Stylesheet Language Transformations (XSLT) and how it can be used to transform XML documents; and
- Annex F – Secure NIEM Exchange Investigation: provides a detailed overview of security labeling, cryptographic binding, and encryption standards examined for use within the CBSA prototype.

2 Operational Architecture & Scenario

This section of the report will provide a high-level overview of the CBSA RADNET operational architecture and scenario. Specifically, this section will examine the following:

- Components;
- Scenario; and
- Sample Data.

2.1 Components

This section of the report will provide a description of the following CBSA RADNET components, as illustrated in Figure 1:

- Marine Ports;
- National Targeting Center (NTC);
- Science & Engineering Directorate (SED); and
- Other Government Departments (OGDs).

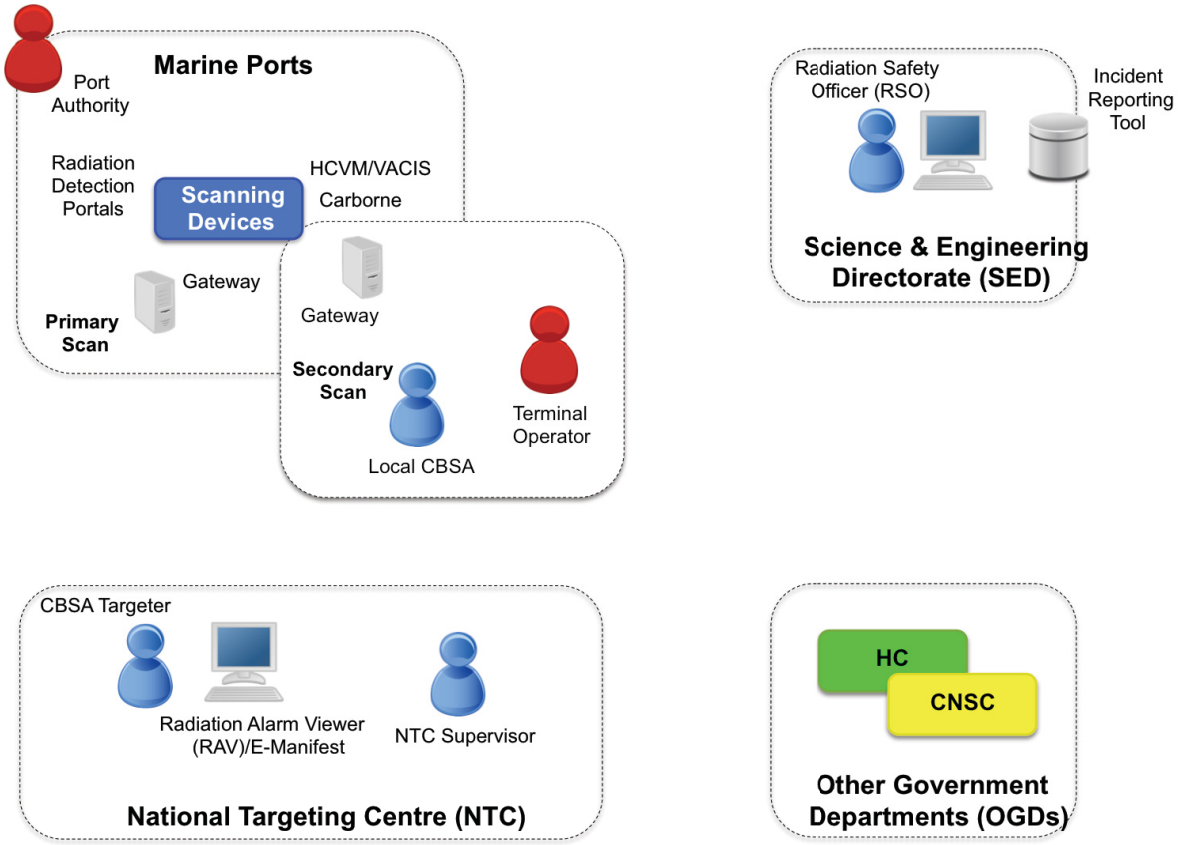

Figure 1 - Components

2.1.1 Marine Ports

Radiation detection equipment is located at Marine Ports in order to prevent dangerous goods from entering the country. Radiation detection portals (illustrated in Figure 2), which are the primary scanning devices, are used to non-intrusively scan all shipping containers arriving at Canadian marine ports. Any shipping containers with an elevated reading, above normal background levels, generate an alarm and undergo a risk assessment and further radiation examination to determine the cause and extent of the radiation. This is done at the NTC. It is worth mentioning that less than 1% of containers trigger an alarm, and, of this 1%, 80% will be cleared quickly as the radiation is naturally occurring. The remaining alarms require a more detailed secondary scan. A secondary scan typically relies on a carborne unit or a HCVM/Vehicle and Cargo Inspection System (VACIS) unit. As the name implies, a carborne unit is one in which the radiation monitoring system is affixed to the roof of a vehicle. Both HCVN/VACIS units are mobile scanners.

Figure 2 - Radiation Detection Portal

2.1.2 National Targeting Centre

CBSA Targeters and NTC Supervisors are responsible for reviewing alarms generated by the radiation detection portal (primary scan). The alerts are received on the Radiation Alarm Viewer (RAV) and compared against the manifest in order to determine whether the radiation is naturally occurring or if further investigation is required. Personnel within the NTC also review advance cargo information prior to the shipment arriving on Canadian soil, or in some cases, prior to leaving the foreign port. Shipments that have been identified as high risk are referred for examination at the first point of arrival or intervention.

2.1.3 Science & Engineering Directorate

Research Safety Officers (RSOs) within the SED are responsible for deciding whether a secondary scan is warranted and whether the shipping container should be quarantined. The scan and manifest information is entered in the Incident Reporting Tool (IRT). RSOs are also responsible for notifying OGDs, if warranted.

2.1.4 Other Government Departments

When shipments with high levels of radiation are detected, CBSA informs both the Canadian Nuclear Safety Commission (CNSC) and Health Canada (HC) in order to prevent radioactive goods from entering Canada and causing health/safety problems to the general public. The CNSC *regulates the use of nuclear energy and materials to protect health, safety, security and the environment and to implement Canada's international commitments on the peaceful use of nuclear energy; and to disseminate objective scientific, technical and regulatory information to the public.*[4] Specifically, the CNSC has developed detailed procedures and identified service standards for the treatment of containers that trigger alarms for man-made radiation. HC is the lead department responsible for coordinating the nuclear emergency response of more than eighteen federal organizations in support of impacted provinces and territories.

2.2 Scenario

The CBSA radiation scenario is based on the RADNet Response Chart found in Annex A and follow-up discussion with CBSA subject matter experts. The CBSA radiation scenario, as illustrated in Figure 3, consists of the following twelve steps:

1. The Radiation Detection Portal (primary scanning device) scans a shipping container at a marine port. The Radiation Portal Monitor (RPM), which records the primary scan in a database and in a file share, identifies that a reading in the scan exceeds a predetermined threshold;
2. The RPM automatically pushes the portal alert event to a database (database-to-database transfer) in the National Targeting Centre (NTC);
3. The CBSA Targeter (3a) views the Portal Alert Event in the Radiation Alarm Viewer (RAV) and compares it with the e-manifest for the shipping container. If the CBSA Targeter does not respond in a predetermined period of time then the Portal Alert Event is automatically sent to the NTC Supervisor (3b) for action;
4. If the Portal Alert Event is at odds with the e-manifest then the CBSA Targeter calls the Radiation Safety Officer (RSO) in the CBSA Science & Engineering Directorate (SED);

[4] http://nuclearsafety.gc.ca/eng/acts-and-regulations/

5. The RSO makes a determination as to whether secondary scanning is warranted for the shipping container. Note that the NTC Targeter also has the authority to request a carborne exam independent of the RSO. In this case the RSO will still need to be contacted to review the spectrum;
6. The CBSA Targeter calls the Local CBSA at the marine port and requests a secondary scan. The Carborne Radiation Detection System (CRDS) is used to perform the secondary scan. As part of the scanning process, the CBSA NTC targeter will contact the Terminal Operator to place a hold on the shipping container;
7. Once the secondary scan has been completed, the Local CBSA will call the RSO and provide notification that the secondary scan results are available for download;
8. The RSO will acquire the secondary scan results. In the case of the carborne scan, the RSO will download the scan from a gateway. In the case of the HCVM/VACIS, scan data is transferred to a workstation via a USB stick. From there it is emailed to the RSO;
9. The RSO, who stores the scan results and images in the Incident Reporting Tool, will examine the secondary scan results and make a determination as to the type of response required;
10. If the carborne scan results are still not conclusive the CBSA RSO will escalate the file to a senior CBSA RSO who will then review the carborne exam results and the manifest data. The senior RSO may then request that an HCVM/VACIS exam is performed;
11. If an HCVM/VACIS exam is requested the results will be reviewed by the Senior RSO and a determination made as to whether the container needs to be forwarded to an OGD; and
12. The RSO will send the scan results (primary scan, secondary scan, image files (HCVM) and templates) via email to the OGDs (CNSC and HC) for incident response.

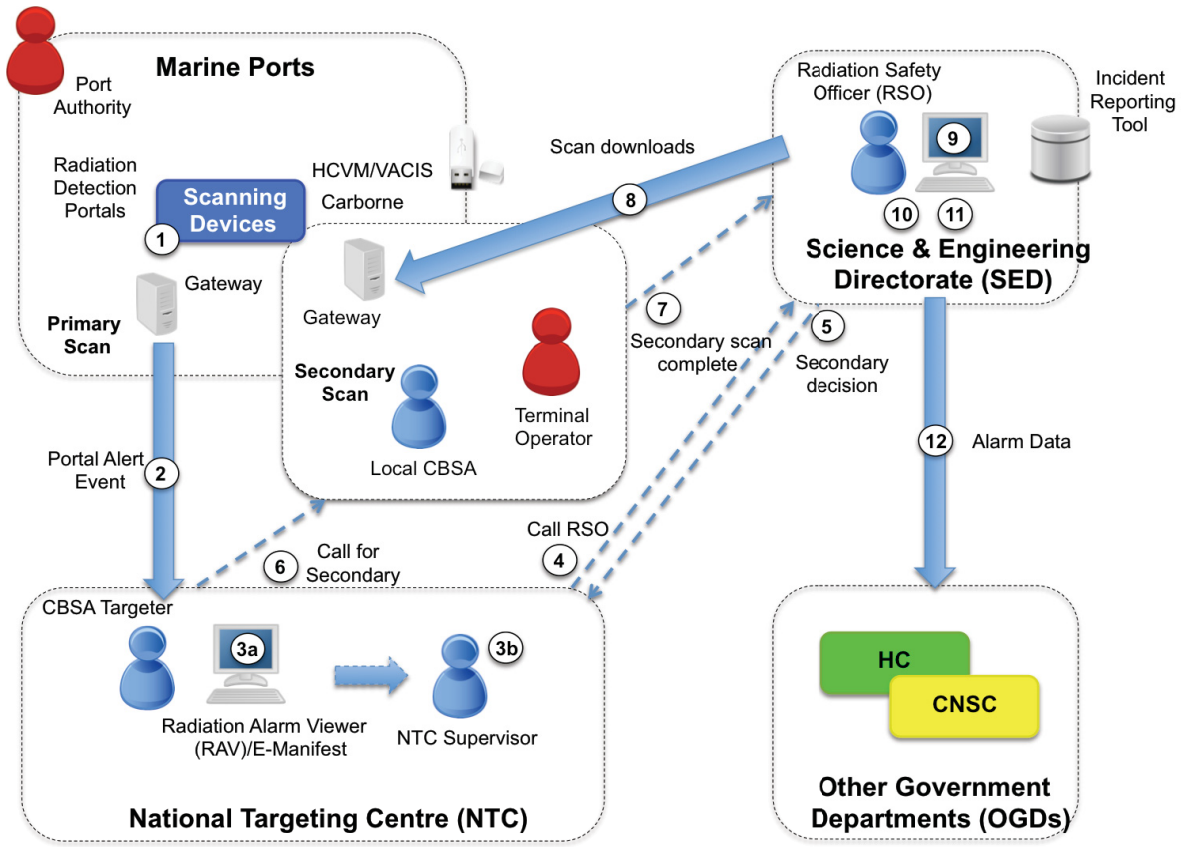

Figure 3 - CBSA Radiation Scenario

2.3 Sample Data

The sample data provided by CBSA is from alarm #184863 which occurred in Halifax in September 2014. This alarm[5] was selected because the shipping container contained stainless steel contaminated with Co-60 during the recycling process. This alarm was eventually sent for a secondary exam using the mobile carborne unit, an HCVM exam to examine the contents, and then referred on to the CNSC for enforcement. The sample data consists of the following:

- Radiation Detection Portal Scan (Primary Scan);
- Alert Data;
- RAV Alert Page;
- Carborne Scan (Secondary Scan);

[5] It should be noted that none of the sample files contain actual numerical data. That data has been either replaced with simulated data or removed.

- HCVM/VACIS Image; and
- Template.

2.3.1 Radiation Detection Portal Scan

The radiation detection portal scan (184863.12n42) is a N.42 XML file that can be viewed either using a text editor or using specialized software that provides a graphical interpretation of its contents.[6] Since the radiation detection portal is responsible for scanning all shipping containers, it is optimized for performance as opposed to detail. Consequently, it has limited capability for identifying specific isotopes. However, despite the fact that the information provided in this scan is limited in nature, it is shared with RSOs and OGDs in order to corroborate measurements from other equipment.

Note – ANSI N42.42

Historically, radiation detection equipment used proprietary data reporting formats. ANSI N42.42 was created to facilitate the sharing of radiation data generated by radiation detection equipment in the field and the scientific personnel required to interpret the data. The standardized format allowed the interpretation of the data to be conducted without having to consider the specific type of equipment used.

2.3.2 Alert Data

IRT 184863.xlsx is a sample of data that is put into the IRT. However, it is also representative of data that is placed in an interface table and sent, using a database-to-database transfer, to the RAV in the NTC.

2.3.3 RAV Alert Page

Alarm Page 184863.jpg is a screen shot of the data that NTC sees on the RAV. It shows the date, time, portal location, radiation data, pictures, and the action to be taken.

2.3.4 Carborne Scan

The carborne scan (Carborne File.n42) is considered the most important file to be shared with the relevant parties as it contains detailed radiation scanning information. As with the primary scan, the carborne scan is in the N.42 data format. Also note that the N42.42

[6] This specialized software is developed in the U.S. and loaned to a number of Canadian departments, including CBSA. The software will display the N.42 files, compare them against a wide range of isotope libraries, and adjust gain to compensate for equipment slightly out of calibration.

file does not contain x-ray image data retrieved from either a VACIS or HCVM. Those would be referenced as separate files.

2.3.5 HCVM/VACIS Image

The HCVM image (HCVM Image.jpg) is the view of the container contents as generated by the HCVM. Higher resolution and clarity images are available from the HCVM that can be manipulated onsite to gain a finer level of detail. However, this file can be considered representative of the type of container contents images that can be obtained.

2.3.6 Template

The template (Form for Containers_Referred_to_the_CNSC_Alarm.docx) is a form that is populated with information and sent, when warranted due to an alarm escalation, to the duty officer at the CNSC. The template can include any information that helps ID the source and aids in response. This can include the following information:

- CBSA assessment as to what the isotope is;
- An indication of whether the radiation type matches the manifest;
- Location of radiation dose coming from the container;
- Manifest information:
    - Shipper;
    - Receiver;
    - Contents; and
    - Weight.

3 High-Level Architecture

The intent of the prototype is not to re-create the CBSA environment in its entirety. Rather, the objective is to prototype a subset of the CBSA environment in order to be able to satisfy the two data-centric security objectives.

The primary objective of this CBSA prototype is the secure exchange of standardized radiation scan data with OGDs. The meaning of "secure" in this context has several implications. Not only will the standardized scan data have an appropriate security label denoting the sensitivity of its content, but the security label will be cryptographically bound to the data using a digital signature. In addition, the digitally signed data will be encrypted to prevent unauthorized disclosure during transit. Lastly, the releasability of the data will be ascertained according to a central security policy.

The secondary objective of the CBSA prototype is to secure scan data, including access to applications containing scan data, within the CBSA prototype environment. Consequently, all scan data will be assigned a security label that will be cryptographically bound to the data using a digital signature. In addition, the digitally signed data will be encrypted in order to prevent compromise in the event that the gateway at the marine port was to be stolen. Lastly, access to an application containing sensitive scan data will be mediated so that only authorized users with the appropriate clearance and need-to-know are able to access the sensitive scan data stored within the application.

To that end, this section will present a high-level architecture for the CBSA prototype environment. It was decided that in order to mitigate risk, the project would be sub-divided into three phases, each of two months duration. Consequently, this section will detail the high-level architecture for each of the following phases:

- Phase 1 – NIEM Exchange;
- Phase 2 – Secure NIEM Exchange; and
- Phase 3 – Policy Mediated NIEM Exchange.

Note – It should be noted that the phases are cumulative. Consequently, the high-level architecture for phase 3 of this project represents the end state or final objective.

3.1 Phase 1 – NIEM Exchange

The high-level architecture for Phase 1 can be seen in Figure 4. This phase simply involves:

1. Retrieval of the scan data from the scan gateway,
2. Sending it to the OGDs in an email, and
3. Conversion of the scan data to N.25 NIEM format.

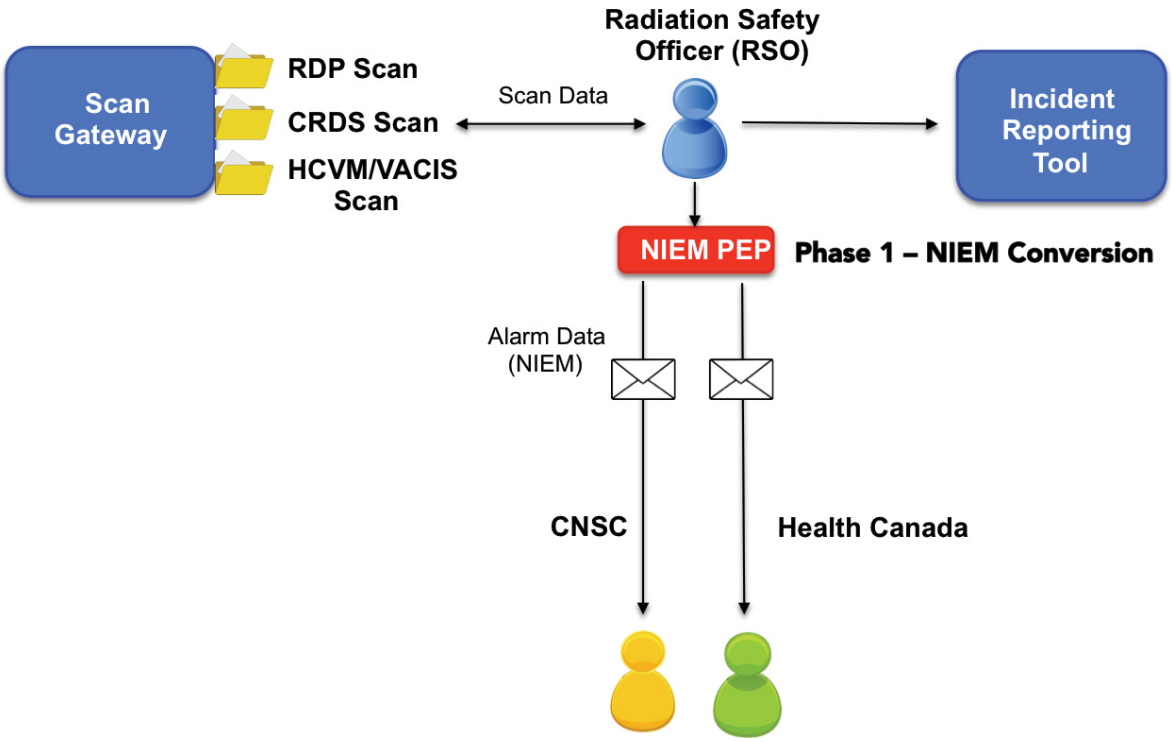

Figure 4 - Phase 1 High-Level Architecture

It should be noted that the IRT component has been moved to a later phase; as agreed to by CBSA at the 29 October 2015 progress review meeting.

3.2 Phase 2 – Secure NIEM Exchange

The high-level architecture for Phase 2 can be seen as Figure 5. This phase involves using DCSS to secure all scan data, including access to scan data, both within CBSA and while in transit to the OGDs.

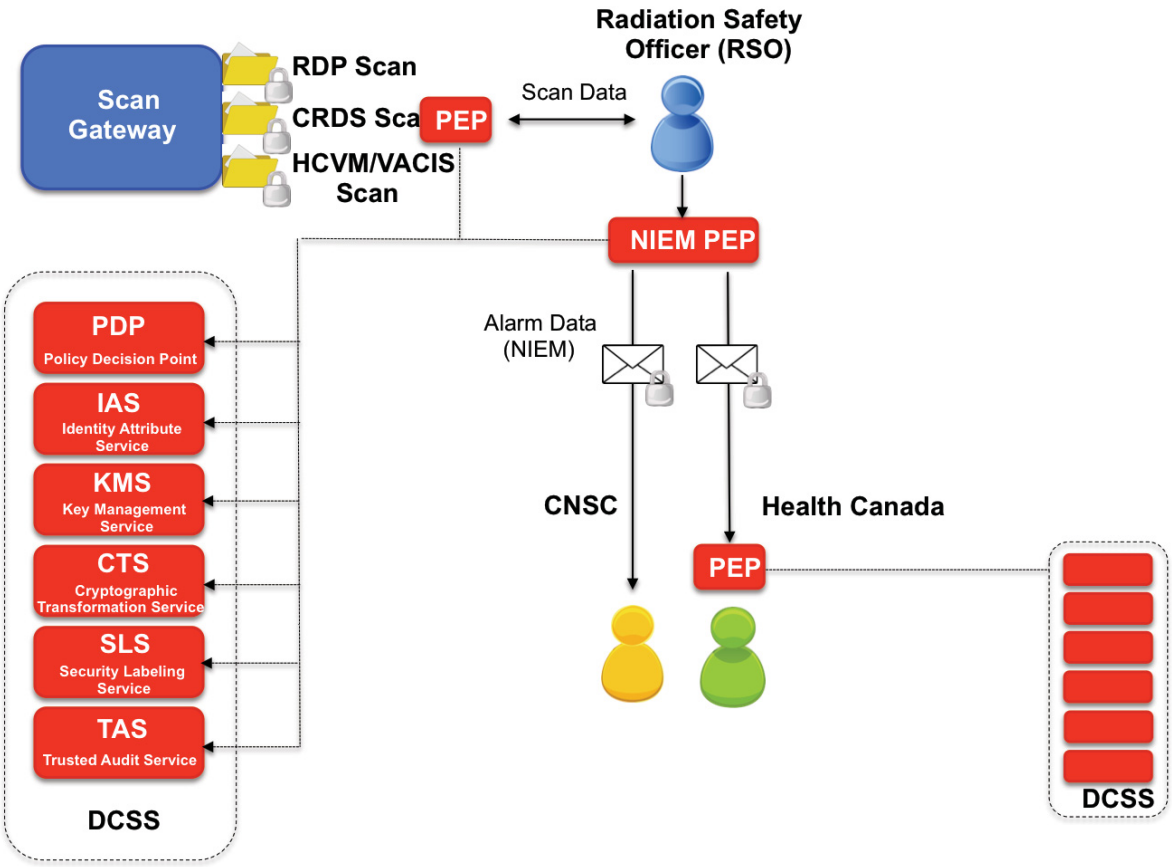

Figure 5 - Phase 2 High-Level Architecture

3.3 Phase 3 – Policy Mediated NIEM Exchange

The high-level architecture for Phase 3 can be seen as Figure 6. This phase is concerned with determining the releasability of the scan data according to the defined organizational security policy. It is also concerned with disseminating the appropriate data to recipients within an organization according to that policy.

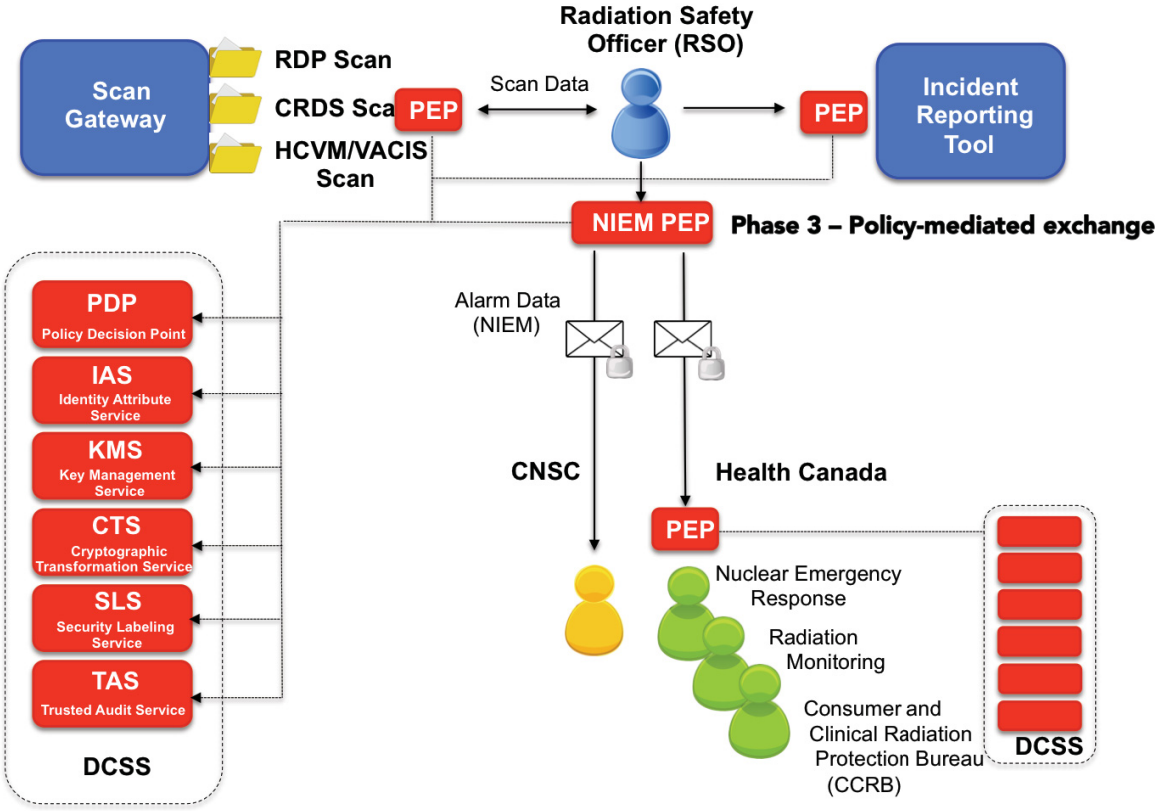

Figure 6 - Phase 3 High-Level Architecture

4 System Architecture

The purpose of this section is to describe the prototype system architecture including the following:

- Virtualization;
- Network Zoning; and
- Components and Services.

Note - The implementation of the prototype will be done in a segregated network environment hosted at Gridway Computing, a third party IT hosting facility.[7] However, it is anticipated that the prototype will be transitioned, if desired, to the CBSA lab facility at the completion of the project.

4.1 Virtualization

The prototype environment has been implemented using VMware virtualization software. The VMware virtualization product family allows a virtual machine (VM) to be isolated from other VMs while using only the required host computing resources. Most importantly, the use of VMs will allow for the seamless transfer of the completed prototype environment from Cord3's lab to CBSA's lab. The following VMware products have been leveraged:

- VMware ESXi Server 5.5; and
- VMware VSphere Client 5.5.

4.2 Network Zoning

As illustrated in Figure 7, the prototype system architecture employs the following IT Network Security Zones:

- Public Zone (PZ);
- Public Access Zone (PAZ);
- Operations Zone (OZ); and
- Restricted Zone (RZ).

[7] Gridway Computing is 100% Canadian with no foreign or U.S. affiliation. The Gridway Data Centre provides a number of safeguards and controls including monitored alarms, security cameras, secure card excess and data centre isolation from the exterior building frame.

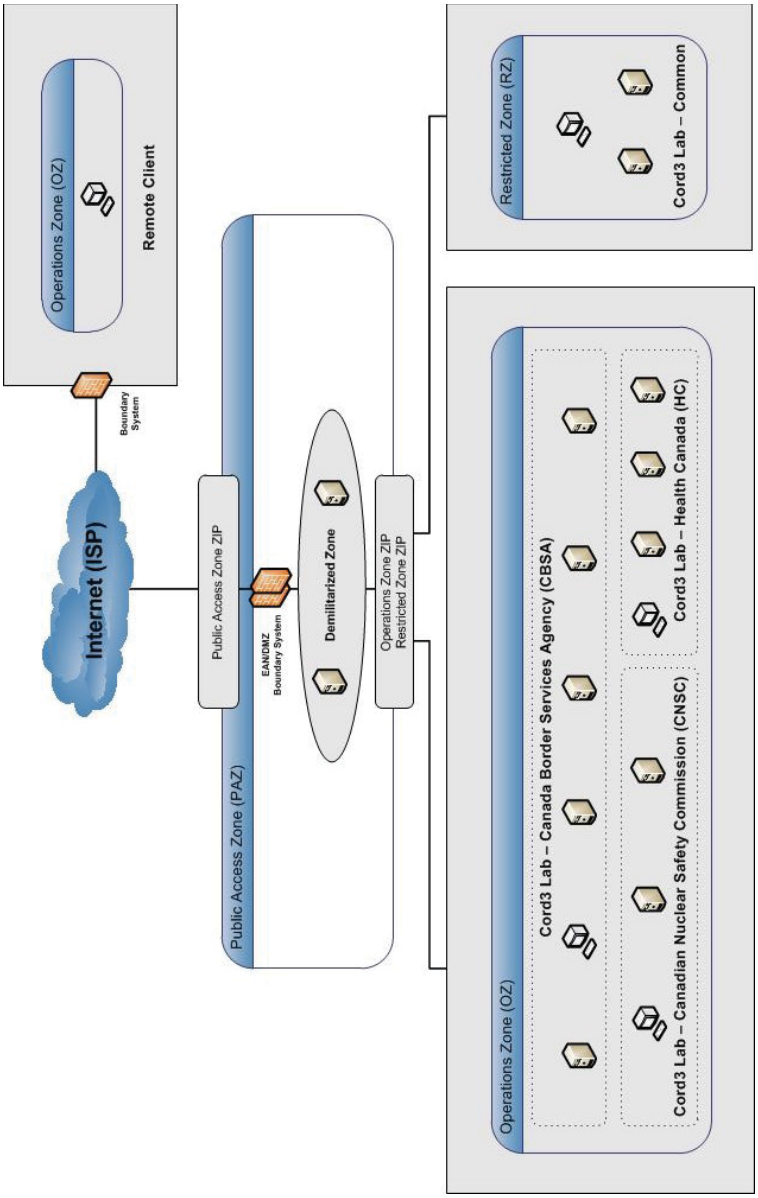

Figure 7 - DCSS CBSA IT Security Zoning

4.2.1 Public Zone

The Public Zone is entirely open and includes public networks such as the public Internet, the public switched telephone network, and other public carrier backbone networks and services. Restrictions and requirements are difficult or impossible to place or enforce on this Zone because it is normally outside the control of the GC as a system owner. The

Public Zone environment is assumed extremely hostile. Any systems delivered in, or interfacing with, the Public Zone should be hardened against attacks.

4.2.2 Public Access Zone

A PAZ mediates access between the prototype networks and the Public Zone. Interfaces to all external services are implemented through a PAZ. In general, sensitive information should not be stored in a PAZ. It may transit or be collected in a PAZ, but it should be transferred to databases in either a RZ or OZ and accessed by applications in the PAZ, thus limiting the amount of sensitive information exposed should a compromise occur.

4.2.3 Operations Zone

The OZ is the standard environment for routine GC operations. It is the environment in which most end-user systems and workgroup servers are installed. All of the prototype systems are implemented in the OZ, albeit in three different domains.

4.2.4 Restricted Zone

Departmental and Internet services network architectures have a restricted zone designed specifically for security services. This zone typically contains IT security related services for the Departmental and Internet services network operations. For the prototype, the RZ will contain Cord3 management and development systems.

4.3 Components and Services

The purpose of this section is to describe the components and services of the prototype system architecture. The prototype virtualization and network architecture are illustrated in Figure 8 and Figure 9 respectively. Specifically, this section will examine the components and services in the following network zones:

- Public Access Zone (PAZ);
- Operations Zone (OZ); and
- Restricted Zone (RZ).

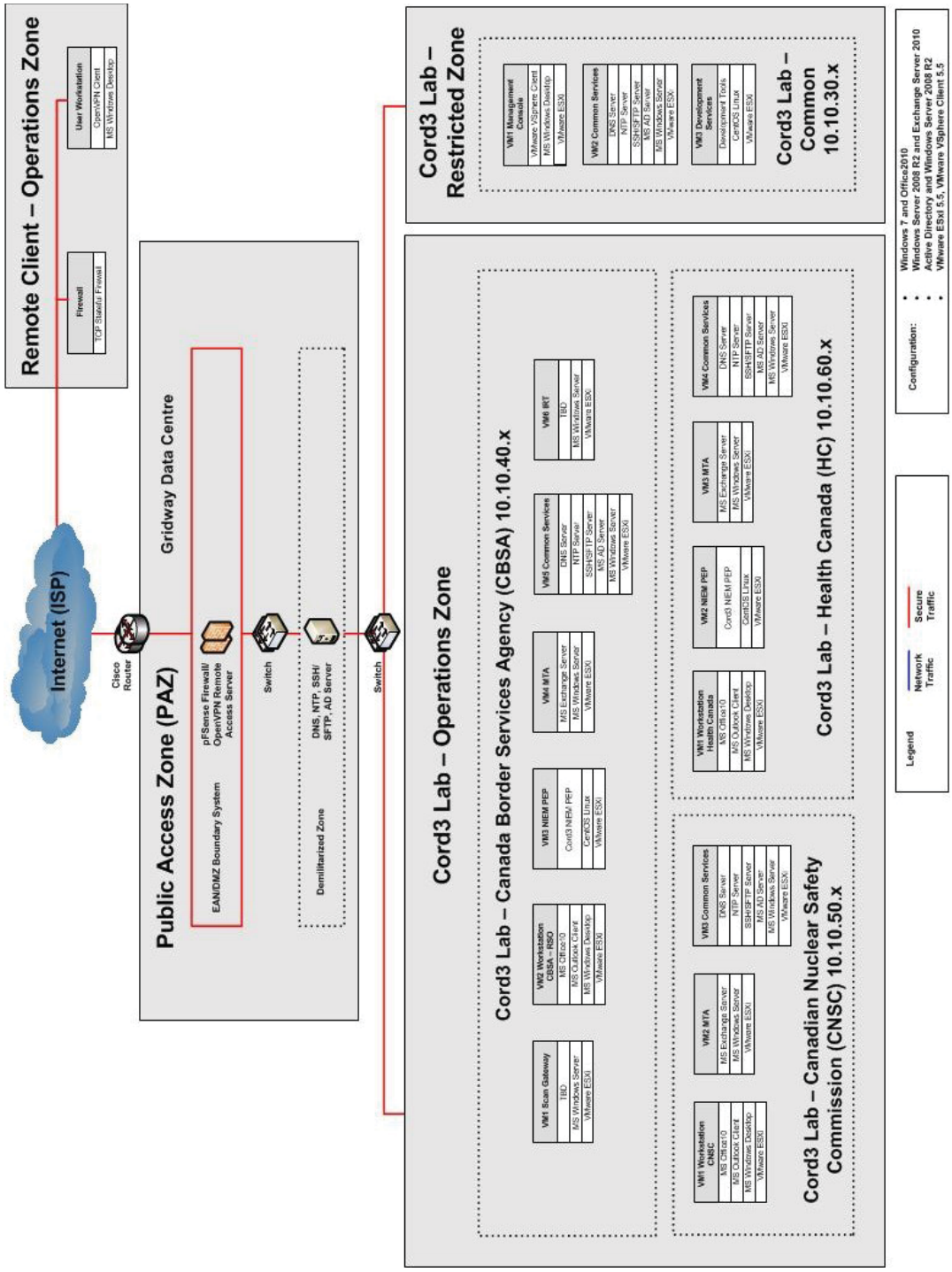

Figure 8 - Prototype Virtualization Architecture

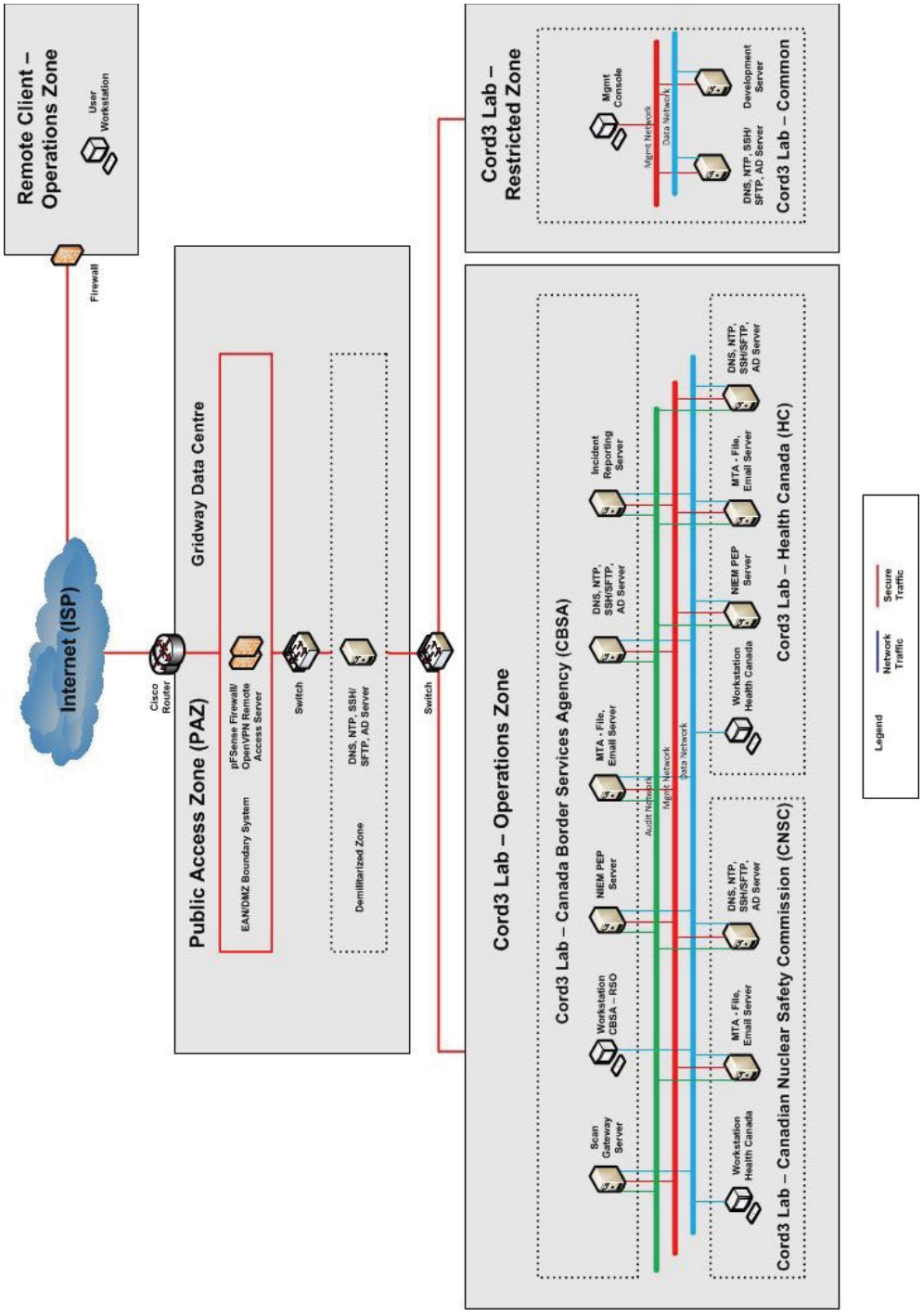

Figure 9 - Prototype Network Architecture

4.3.1 Public Access Zone

The prototype environment includes pfSense as the EAN boundary system. pfSense provides stateful firewall services and acts as a gateway to enforce a boundary between

the prototype environment and remote users. OpenVPN is an open-source software application that implements virtual private network techniques for creating secure point-to-point or site-to-site connections in routed or bridged configurations and remote access facilities.

Table 1 - EAN Components

Host	Service	Description
Cisco Router	Routes traffic from Internet	Routing and Network Access Control.
	Switch Ethernet frames between ports	Switches work at Layer 2 of the OSI model to switch Ethernet frames between ports.
EAN DMZ Boundary System	pfSense Firewall Service	• Access Filter • Activity Logs • Traffic Logs • TCP Stateful Filter
	OpenVPN Service	• IPSec VPN

Table 2 - DMZ Components

Host	Service	Description
Common Support Services	DNS/Bind Service	Common services DNS/Bind.
	NTP Service	Common services NTP.
	SSH Service	A service that provides strong authentication and secure communications over insecure channels.
	SFTP Server	A service to permit secure data interchange.

4.3.2 Operations Zone

The operations zone contains the three network domains; CBSA, CNSC and HC.

Table 3 - CBSA Components

Host	Service	Description
Scan Gateway	Microsoft Windows Server 2008 R2	The scan gateway system contains the network shares used to store the scan data.
RSO Workstation	Microsoft Windows 7 and Microsoft Office 10 (including Microsoft Outlook Client)	The RSO workstation is meant to emulate the workstations used by RSOs to retrieve scan data, send emails to OGDs, and connect to the IRT.
NIEM Policy Enforcement Point (PEP)	CentOS Linux and NIEM PEP	The NIEM PEP serves to convert scan data sent by the RSO to N.25 NIEM formatted data. It also performs the releasability check on the data to be exchanged.

Host	Service	Description
MTA Server	Microsoft Windows Server 2008 R2 and Microsoft Exchange Server 2010	The MTA will serve as the mail server for the CBSA domain. Specifically, it will send CBSA email to the CNSC and HC domains.
Common Support Services	Microsoft Windows Server 2008 R2 with Active Directory	The AD server will serve as the Primary Domain Controller (PDC) for the CBSA domain, including providing authentication, DNS, and NTP services.
DCSS	CentOS Linux and DCSS	The CBSA DCSS provides the requisite security services to enable data-centric security within the CBSA domain. The DCSS will be implemented in the CBSA domain in Phase 2.

Table 4 - CNSC Components

Host	Service	Description
CNSC Workstation	Microsoft Windows 7 and Microsoft Office 10 (including Microsoft Outlook Client)	The CNSC workstation is meant to emulate the workstation used by the CNSC Point of Contact (POC) to receive radiation emails from CBSA.
MTA Server	Microsoft Windows Server 2008 R2 and Microsoft Exchange Server 2010	The MTA will serve as the mail server for the CNSC domain. Specifically, it will receive CBSA emails sent to the CNSC domain.
Common Support Services	Microsoft Windows Server 2008 R2 with Active Directory	The AD server will serve as the Primary Domain Controller (PDC) for the CNSC domain, including providing authentication, DNS, and NTP services.

Table 5 - HC Components

Host	Service	Description
HC Workstation	Microsoft Windows 7 and Microsoft Office 10 (including Microsoft Outlook Client)	The HC workstation is meant to emulate the workstation used by the HC Point of Contact (POC) to receive radiation emails from CBSA.
MTA Server	Microsoft Windows Server 2008 R2 and Microsoft Exchange Server 2010	The MTA will serve as the mail server for the HC domain. Specifically, it will receive CBSA emails sent to the HC domain.

Host	Service	Description
Common Support Services	Microsoft Windows Server 2008 R2 with Active Directory	The AD server will serve as the Primary Domain Controller (PDC) for the HC domain, including providing authentication, DNS, and NTP services.
NIEM Policy Enforcement Point (PEP)	CentOS Linux and NIEM PEP	The HC NIEM PEP serves to convert scan data sent by the RSO back to its original format, before being stored on the File Server. It also is responsible for creating a new email containing links to the files on the File Server. The NIEM PEP will be implemented in the HC domain in Phase 2.
DCSS	CentOS Linux and DCSS	The HC DCSS provides the requisite security services to enable data-centric security within the HC domain. The DCSS will be implemented in the HC domain in Phase 3.
File Server	Microsoft Windows Server 2008 R2	The file server contains the network shares used to store the scan data within the HC domain.

4.3.3 Restricted Zone

Table 6 - Common Components

Host	Service	Description
Management Console	System Console Service	Provides management console access.
Development Server	CentOS Linux and Development Tools	Cord 3 Development environment.
Common Support Services	Microsoft Windows Server 2008 R2 with Active Directory	The AD server will serve as the Primary Domain Controller (PDC) for the management domain, including providing authentication, DNS, and NTP services.

5 Phase 1 – NIEM Exchange

This section of the report will document Phase 1 of the CBSA prototype. Specifically, this section will examine the following:

- NIEM Policy Enforcement Point (PEP);
- EXtensible Stylesheet Language (XSLT) Translation;
- Testing Summary; and
- IRT Proposal.

5.1 NIEM PEP

The NIEM PEP is responsible for intercepting email sent between CBSA and OGDs, extracting the radiation scan data, transforming the radiation scan data, and then composing new email messages to be sent to the intended recipients with the transformed radiation scan data. Specifically, the NIEM PEP performs the following operations:

- When an email is sent from the CBSA Exchange Server to the HC Exchange Server or the CNSC Exchange Server, the Exchange Smart Relay will route the email through the NIEM PEP;
- All email can only be sent over port 25 so the NIEM PEP iptables will forward connections on 25 (privileged port) to 10025 (unprivileged port);
- The proxsmtp intercept will listen on 10025 and will route email to an MTA on the NIEM PEP through port 25;
- When an email is processed through, the proxsmtp intercept, it will store the email to a temporary file at /usr/local/var/email and sets an environment variable EMAIL to this file;
- The proxsmtp intercept will then process this file: take a copy of the original message and walkthrough the incoming message to extract all the attachments; and
- The XSLT transformation is now performed on the attached message elements.

5.2 XSLT Transformation

As mentioned previously, the radiation scan data consists of two N.42 documents, an HCVM image file, and a Word document. All of this data will need to be transformed as per the N.25 Information Exchange Package Documentation (IEPD). However, there are a number of impediments to easily achieving this transformation. These impediments include the following:

- Radiation detector manufacturers have implemented the N.42 schema autonomously, therefore mappings via an XSLT Transformation from N.42 to N.25 and back again cannot be guaranteed;
- At this point, not all radiation detector manufacturers use N.42;
- The manufactures that have implemented N.42, have not implemented it consistently enough to allow an N.42 mapping to N.25 for one manufacturer's detectors to work with another manufacturer; and
- The N.25 IEPD profile contains similar information found in N.42, but a mapping between N.42 schema and N.25 schema does not currently exist.

This section will examine the following options in terms of overcoming these impediments and achieving the requisite data transformation:

- Embed N.42 documents;
- Map N.42 documents; and
- Leverage existing mapping.

5.2.1 Embed N.42 Documents

The N.25 *RadiationDeviceMessage* is designed to allow non-XML files to be included in the message via metadata elements recording details of the included file and an encoded (e.g. base64, uuencode, etc.) element. Consequently, as a first step the two N.42 documents, the Word document and the image file were embedded in the N.25 *RadiationDeviceMessage*. This was accomplished in Phase 1 of the prototype and used for the remainder of the prototype.

5.2.2 Map N.42 Documents

The N.42 schema to N.25 schema mapping operations leverage the following key XML schemas:

- CBRN: the XML elements of interest to the CBRN working group of NIEM;
- N.25 IEPD profile of CBRN: define a set of messages about radiation devices, measurements and alarms (version 2.1);
- ANSI N.42-2006: detail an output format for radiation detectors, intended to provide commonality amongst vendors; and
- ANSI N.42-2012: an update to ANSI N.42-2006.

At this point in time, the N.25 IEPD profile contains similar information found in N.42, but a mapping between N.42 and N.25 has not yet been released and some discrepancies must be resolved, for example:

- Elements describing something may have different names;

- Elements may have different structures;
- Content may be carried by attributes in one standard and by elements in another;
- Type of an element may be different between standards even when the element names match; and
- Elements in one standard could require values to be drawn from an enumerated list that is not identical to the other.

5.2.3 Leverage Existing Mapping

At this point in time, off-the-shelf mappings do not exist for the XSLT transformation between the N.42 schema (2012) and N.25 schema (version 3.1). It is expected that radiation detector manufactures will eventually adopt the current N.42 schema standards thereby ensuring future alignment. Until this mapping has been completed, this level of transformation cannot be achieved.

5.3 Testing Summary

This section describes the prototype test plan and results. Complete testing information for Phase 1 can be found in Annex B. The testing during Phase 1 shall be in support of a NIEM information exchange between CBSA and OGDs. The principal target of the experiment using CBSA radiation data, shall address the ability of the CBSA to send an email with file attachments, to OGDs, using a NIEM PEP to apply the NIEM transformation. Figure 10 illustrates the relationship between the CBSA and OGDs. Specifically, this section will describe the following aspects of Phase 1 testing:

- Testing Overview;
- Types of Tests; and
- Test Results.

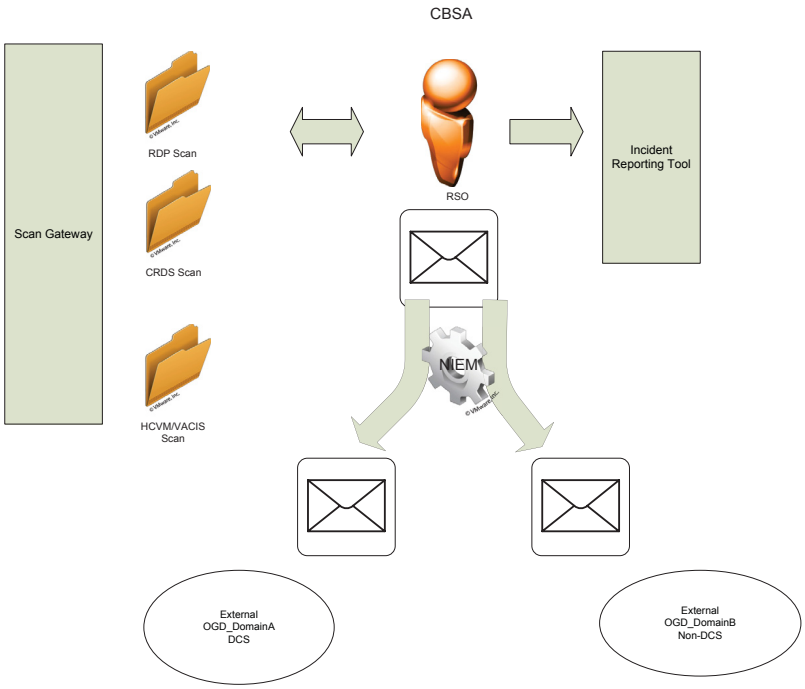

Figure 10 - CBSA Data Transfer to OGDs

5.3.1 Testing Overview

The OGDs will be sent

1) two N.42 files:
   a) one from the radiation detection portal (primary scan) and
   b) one from the carborne system (secondary scan),
2) one image file from either the VACIS or HCVM and
3) a Microsoft Word document.

Only a unidirectional transmission of email will be tested from the originating RSO to the receiving OGD.

5.3.2 Types of Tests

In Phase 1 there will be two types of tests:

- XML Tests – the verification that the XSLT transforms have been carried out correctly and accurately; and

- Email Tests – the functional tests to ensure that the N.42 data can be sent, as NIEM structured data from the CBSA to OGDs using email as the transport mechanism.

The XML tests, the XSLT transforming of the N.42 to N.25 format, will be carried out in a layered approach consisting of data quality, and operational quality. Data quality will address individual components. Operational quality will address the inputs and outputs of the model.

The following tests will be carried out:

- Automated validation checks to assess objective data and semantic quality;
- XMLSpy validation of distribution schemas to test for XML Schema conformance; and
- Xerces validation of a specially generated instance that represents every component from the model to test for XML Schema conformance.

These are a series of tests designed to ensure that the "system under test" can send an Email with the radiation scan files and images as attachments from the CBSA to OGDs in the NIEM structure.

5.3.3 Test Results

As can be seen in Annex B, the Phase 1 prototype successfully passed all five testing scenarios.

5.4 IRT Proposal

Based on the current understanding of the Incident Reporting Tool (IRT), the project team has developed a proposal for an IRT simulator that provides a reasonable approximation of how information assets are managed in the context of incident reporting. While there is an overarching objective to make the demonstration of data-centric security as relevant to existing CBSA practices, the primary goal of this effort is to show how data-centric security can be used to exert control over organization assets while enabling the exchange of assets with other organizations subject to policy. It is expected, therefore, that the creation of an IRT simulator will show the application of data-centric security on organization assets without the need to have direct access to the current IRT software. Alignment between the simulator and the actual IRT solution will be ensured by using, where possible, the database, protocols and data service solutions that are currently in use by CBSA in general and the IRT in particular.

The IRT is used by Radiation Safety Officers (RSOs) to record information, specifically manifest and radiation data, pertaining to radiation incidents. RSOs access the IRT, which has a client-server architecture with a database back-end, using username and password credentials. However, all RSOs have the same view within the IRT. Data typically found in the IRT can be seen in the sample data file (IRT 184863.xlsx) provided by CBSA.

The IRT is also used by students for data entry purposes. Specifically, the students are hired to enter manifest and radiation information for containers that have not been flagged for additional screening. As it currently stands, within the IRT application there is *no way to control what data the students have access to*. Consequently, sensitive incident data stored in the IRT application, and specifically the database, is at risk of unauthorized disclosure. CBSA would like to implement an access control solution that limits student access to data entry and prevents them from accessing potentially sensitive information.

The intent is to emulate the IRT application by developing a prototype application consisting of a web server and a database. The web server would host the presentation and application logic, whereas the database would store the manifest and radiation data retained by the application. A database Policy Enforcement Point (PEP) would be developed in order to work with Data-Centric Security Services (DCSS) to secure sensitive data within the IRT application and control user (RSO and student) access to this data. This architecture can be seen in Figure 11.

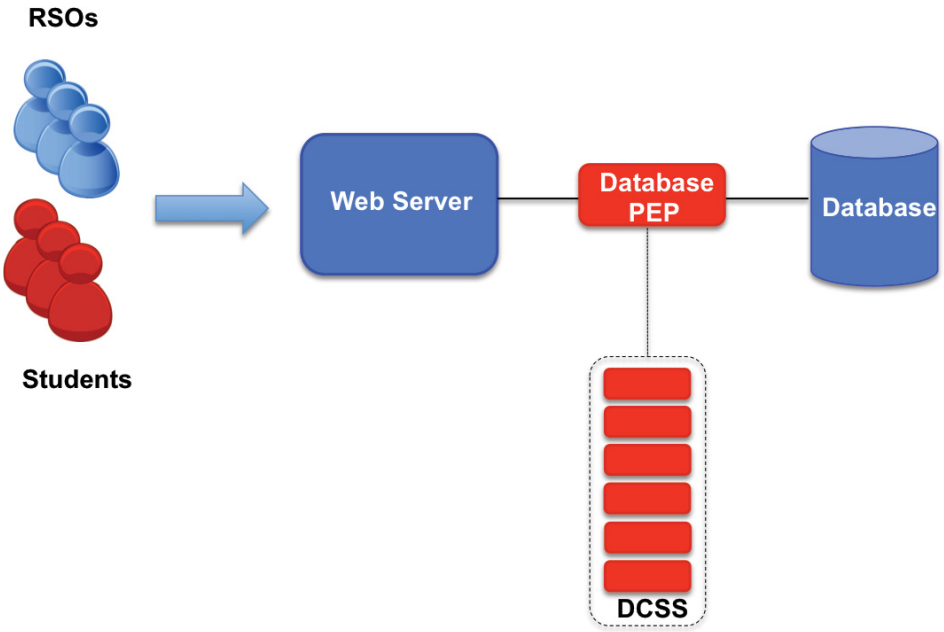

Figure 11 - Proposed IRT Architecture

It is envisioned that a number of additional fields would need to be added to the database schema in order to accommodate security metadata. This security metadata would be used to enable both column and row access control. For example, in Figure 12, a database table contains both columns and rows that are sensitive and require enhanced privilege to view. With the data traversing the DCSS PEP, the information that is returned to a user submitting a query would be vetted so that it is in accordance with the current security policy and the user's current policy rights. Furthermore, it is envisioned that sensitive fields within the database would be encrypted, while leaving the rest of the data unencrypted in order to facilitate database searches.

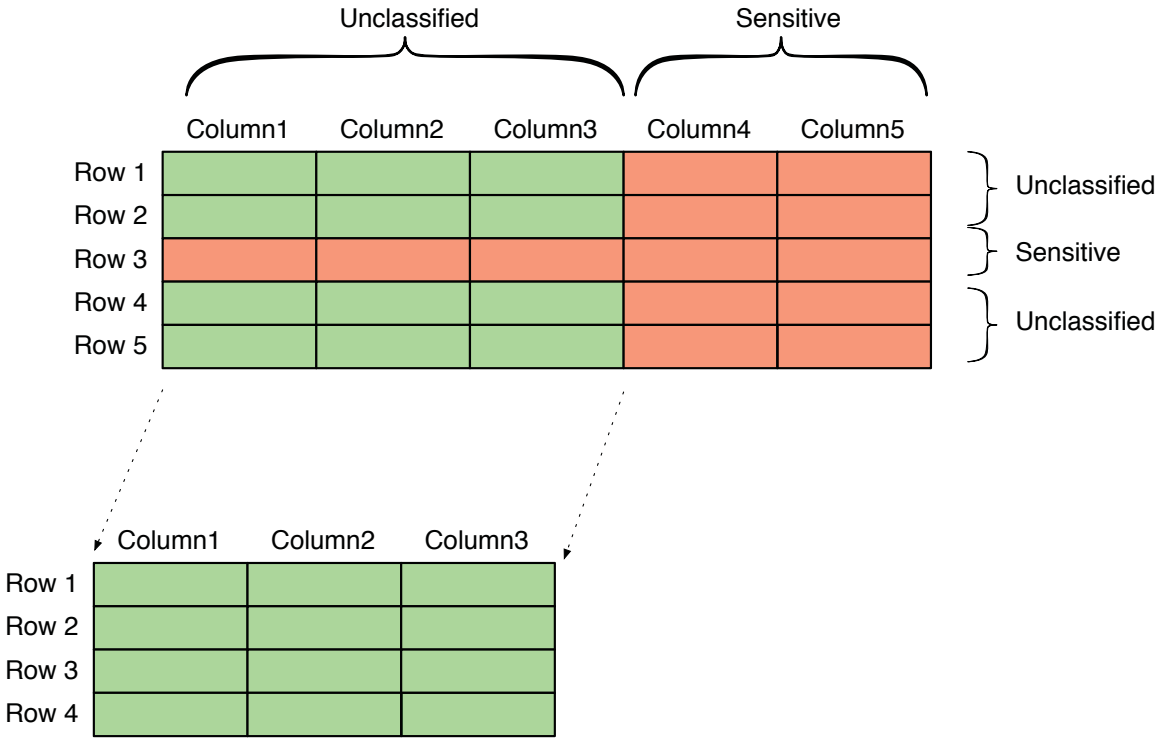

Figure 12 - Proposed IRT Database

6 Phase 2 – Secure NIEM Exchange

This section of the report will document Phase 2 of the CBSA prototype. Specifically, this section will examine the following:

- Standardized NIEM Exchange Investigation;
- Secure NIEM Exchange Investigation;
- Phase 2 Prototype;
- Testing Summary; and
- Additional IRT Design Detail.

6.1 Standardized NIEM Exchange Investigation

Note - In order to have a proper appreciation, and comprehension, of the following material, it is recommended that readers first review the XSLT Primer (Annex E).

As part of its response to an anomalous radiation detection event, CBSA bundles up a set of documents for further investigation. The documents could include scan data from one or more radiation detectors, one or more images, and an analysis document. Currently, the documents are bundled together as email attachments but there is interest in exploring a more standards-based way of handling the bundling.

One interesting candidate for document bundling is NIEM, and in particular the N.25 IEPD. Reasons for selecting the NIEM N.25 IEPD (henceforth N.25) include a departmental directive to look into this option but there are other practical benefits. N.25 handles document aggregation, including documents in any format (images, raw radiation scan data, etc.) and by doing so creates a more reliable and verifiable transport for data than email attachments. N.25 is also designed for the CBRN space and has data elements to contain the information unique to that space. A transport like N.25 also facilitates document exchange with other organizations and especially with other automated systems. Of course a data format like N.25 also allows for security labelling to support a data centric security architecture.

6.1.1 NIEM, N.25, and Related Standards

NIEM consists of a core set of data elements that describe data types common to most information exchanges including things like addressing, location, and identification for example. In addition, NIEM also supports a number of domain specific data elements unique to particular organization or industries. Of particular interest is the CBRN domain which builds on the NIEM core and adds data elements for the CBRN space.

NIEM is not necessarily intended to be used as is. The NIEM (including CBRN) data elements are intended to be packaged up into messages of the sort that would be exchanged between corresponding organizations or systems to support domain activities. The N.25 IEPD defines about 50 messages that allow for details about radiation scanners, detection events, radiation alarms, and so on to be exchanged.

There are also related standards that are of interest to CBSA. In particular, the ANSI (American National Standards Institute) N42 series includes a standard covering the output of radiation detectors (N42.42). N42 was created to facilitate consumption of radiation detector data by providing a common and non-proprietary data format that all detectors could use.

The standards all exist in various versions and it is important to understand the relationships between them in order to understand the scope of the problem at hand. ANSI released a version of N42.42 in 2006 that was used as detector output by some manufacturers.

The N.25 IPED is based on NIEM v2.1 (core and CBRN domain) and dates back to 2009. More specifically, the IEPD is N.25 v1.1.40. This ANSI and NIEM CBRN work was done in parallel so although the two groups produced standards covering much of the same ground (a least insofar as detector data goes – N42 is more focused than NIEM CBRN), the data models and syntax were not compatible. At some point, ANSI and NIEM realized that aligning their work would best serve their target communities. This was done and the ANSI N42 series released in 2012 included the newly aligned data models. The corresponding NIEM update was released in 2013 as v3.0 (core and CBRN). We can see how well the data models were aligned by looking at Figure 13.

Figure 13 - N42.42 (2012) to NIEM v3.0 CBRN Mapping

A portion of the N42.42 (2012) data model is shown on the left of the image and a portion of NIEM v3.0 is shown on the right. The lines connecting the two data models were added by a software tool that facilitates mapping between different data models. In this instance, the tool was able to figure out virtually all of the mappings between N42.42 (2012) and NIEM v3.0 without any manual intervention. Unmapped elements have trivial mappings and could also be automated. The implication of this alignment is that it should be quite easy to take data in N42.42 (2012) format and convert it into NIEM v3.0 format. Although ANSI N42 and NIEM are now aligned, the N.25 IEPD messaging format has yet to be updated to include the modified data models. N.25 was supposed to have been updated in 2014 as N.25 v3.0.54 (based on NIEM v3.0 core and CBRN) but the work has been delayed with no new release date in sight. The importance of this work can be seen in Figure 14.

Figure 14 - N42.42 (2012) to N.25 v1.1.40 Mapping

In the above image, with N42.42 (2012) on the left and N.25 v1.1.40 on the right, all the mappings had to be created manually and many elements still require mapping. Even worse, it may not be possible to map some of the elements. For details about the sort of mapping problems that can occur, see the XSLT Primer (Annex E). Until N.25 is updated, it will be difficult at best to properly map N42.42 data into N.25 messages. Quite apart from the current difficulty with an out-of-date N.25, CBSA must deal with data formats other than N42.42 (2012). Not all detector manufacturers have upgraded from the older N42.42 format and CBSA also uses devices that have their own data formats.

6.1.2 Prototype Delivery

For the project, the objective was to support the aggregation of a JPEG image file, an MS-Word document, a detector scan in N42.42(2006) format, and a scan in N42.42(2012) format. N.25 allows for any kind of document to be bundled in an N.25 message as encoded binary data. This capability was leveraged for the documents that would have to have been carried as binary data in any case (the image, Word, and N42.42(2006) documents), and extended this to include the N42.42(2012) scan data as well.

In effect, an N.25 message document was created without doing any mapping at all. This was not a particularly compelling use of N.25 but at least it did allow the aggregated documents to be carried with perfect fidelity.

6.1.3 Additional Scan Data

CBSA has provided a total of nine data samples from a variety of instruments. They are as follows:

- Radiation Portal;
- HCVM (Carborne);
- Digital Density Meter – Alcohol;
- Elemental Analysis – CHN;
- Fumigant Analysis auto Vikane Cal ;
- GCFID – Congeners;
- Ion Chromatography – Acid;
- IRMS – Alcohol; and
- Sugar Level.

Each sample is in one of 3 data formats:

- XML;
- CSV; and
- PDF.

The first step in any data analysis leading up to a data mapping or data transformation is to understand what we know and don't know about the data. The second step is to understand what we know and don't know about the environment in which any data mapping or data transformation will exist and operate. The information gathered during this step will help determine factors like:

- The transformations required;
- The technologies (standards, software, protocols, etc.) that may constrain the solution; and
- How the transformation and technologies will support the overall project goals.

6.1.3.1 Data Analysis

A complete data analysis can be time consuming and difficult. If sample data is in hand, we need to see if the samples expose all possible data elements. With a complete inventory of data elements available, we must then see if the samples tell us enough about data types and data ranges to create a good data transform. If not, we need to see if any supporting documentation is available that might help in building a robust

transform. The goal is to avoid developing a data map or data transform that is fragile and prone to breakage as unexpected data or values present in the input.

At the same time as we are doing the data analysis, we need to investigate the data consumers, whether human or machine. For example, if the data consumers are only interested in output values, we may be able to ignore any data elements associated with scanner calibration. This can simplify both the data analysis and ultimate data transformation and therefore reduce the cost and complexity of a solution. Of course it is assumed that the data consumers have already been identified as a necessary precursor to the data analysis.

The format in which the data samples exist can help or hinder the data analysis and ensuring data transformation. We have data in three formats, so the next few subsections discuss each in turn.

6.1.3.2 Comma Separated Values (CSV)

CSV is the lowest common denominator format for data exchange and is often used to expose data in a proprietary internal format. At best CSV exposes data in a machine readable format that may be useful to other tools although simply exporting data in CSV format does not imply that other tools will be able to make sense of the data.

CSV data can represent simple tabular data. This limitation may mean that the CSV format may hide a rich internal data format that contains structural relationships and other semantic information. For example, a tool that used a relational database store with a highly normalized data model would have to denormalize the data completely to present it in CSV format. In the process, data hierarchy would be lost which could affect the quality of any resultant data transform.

It is also worth mentioning that the CSV data format does not stand alone. Out of band details must be communicated along with the CSV encoded data in order to fully interpret the data. For example, the character encoding (e.g., ASCII, UTF-8, 8859-12, Code page 1252) used in the CSV data must be known or the data values may turn into gibberish when used by a data consumer. Similarly, an ostensibly CSV formatted file may in fact use a delimiter other than a comma. Indeed, one of our sample data files uses tab delimiters instead of commas and is imported into Excel incorrectly without manual intervention.

6.1.3.3 Portable Document Format (PDF)

The PDF format actually supports some of the data exchange goals of the current project. PDF allows data and images to be mixed in the same document which roughly

corresponds to our need to create a package of data based on several different types of input. PDF documents can also be digitally signed which address security concerns. However, PDF is primarily a presentation format and as such, its strength is not in machine oriented data exchange even though PDF can be parsed by machine.

Obviously, PDF is not N.25, so any requirement to use N.25 is not satisfied. More importantly though, PDF is intended to support paginated output for human consumption. As such, PDF output typically contains a great deal of purely presentational details like page headers and footers that obscure the data. The data itself may also be presented with accompanying natural language commentary that is impossible to mine for semantic details as it might relate to the actual data. Of course the data may be presented in summary formats (e.g., graphs, charts) that hide the raw data. In short, PDF makes for a very poor data exchange format, especially for automated processes.

6.1.3.4 eXtensible Markup Language (XML)

When data is available in XML format, it is likely that XML is the native data storage format of a piece of equipment. XML therefore likely captures the entire data set producible by the equipment. Furthermore, if the XML is defined by a data model (e.g., XML Schema, Schematron, Relax-NG), complete data range and type information may be available.

Even if XML is not the native data storage format, XML is rich enough to expose the full richness of most proprietary or database formats. Still, we need to base our data analysis on more than single data samples as the data samples may expose “drift” between the data model and actual data instances. Of course, XML is no more vulnerable to this than any other format.

6.1.3.5 Environmental Analysis

An environmental analysis is all about looking at a data mapping or data transform in a broader context. It is not enough to say “map everything to N.25” and assume that all will be well. The stakeholders who create data to be transformed and the stakeholders who consume it (external or internal) will all have input into a data transformation project.

Data transformation may also be a part of a larger business process reengineering effort which by its very nature requires a more comprehensive view than would be suggested by the data transformation alone.

Of course, project scope, budget, and timelines will affect the level of effort that can go into developing a data transformation. The reverse is also true in that a desired data transformation may impose budgetary and scheduling requirements.

6.1.3.6 Conclusions

Except for trivial cases, data analysis is not simple and exists in a context established by business and stakeholder constraints. Sorting out the business details and working with stakeholders may require a significant project management effort and may in fact represent the bulk of the work in a data transformation project.

6.2 Secure NIEM Exchange Investigation

This section will summarize the results of an investigation into the means with which to secure the NIEM exchange. This is discussed in additional detail in Annex F. The U.S. Government, the North Atlantic Treaty Organization (NATO), and NIEM are active in terms of developing information exchange security standards. These include security labeling, cryptographic bindings, and encryption. Specifically, this section will look at U.S government, NATO and NIEM standards in these areas, as well as justifying the approach taken within the CBSA prototype.

6.2.1 U.S. Government Standards

Within the U.S. Government the Director of National Intelligence has a number of data specifications related to security labeling, binding and encryption. These specifications include the following:

- Need To Know Metadata (IC-NTK);
- Trusted Data Format (IC-TDF); and
- Information Security Marking Metadata (IC-ISM).

IC-NTK metadata facilitates automated systems making a “need-to-know” (NTK) access determination about an information resource. These metadata are used to represent the system-specific properties assigned to an information resource that will be used, in conjunction with information about the user, and possibly other information, to determine the user’s access to the data.

The IC-TDF defines detailed implementation guidance for using XML to encode IC-TDF data. This is the IC format for binding assertion metadata with data resource(s). This TDF functionality supports the IC way ahead strategy of implementing secure cloud-based

information exchange and discovery on the IC Enterprise. An example of the XML elements within this Data Encoding Standard are EncryptionInformationGroup, KeyAccessType, EncryptionMethodType, and BindingType(SignatureValue), etc.

The IC-ISM enterprise data encoding specification defines XML elements and attributes, associated structures and relationships, mandatory and cardinality requirements, permissible values, and constraint rules for representing electronic information security markings. The standard supports Executive Order (EO) 13526, Classified National Security Information which "prescribes a uniform system for classifying, safeguarding, and declassifying national security information", across national security disciplines, networks, services, and data.

6.2.2 NATO Standards

Within NATO there are two specific standards, which are under development, and are relevant to the securing of information exchanges; STANAG 4774 Confidentiality Metadata Label Syntax and STANAG 4778 Metadata Binding Mechanism.

The NATO STANAG ADatP-4774 Confidentiality Metadata Label Syntax, provides common XML-based formats and syntax for security policies and confidentiality metadata. Information objects and data assets can be labelled to support access and release decisions in a manner that is understandable to all partners.

The objective of NATO STANAG ADatP4778 Metadata Binding Mechanism is to describe generic concepts for binding of metadata to data objects for use within a specific domain or enterprise, and that can easily be interpreted and processed among a federation of enterprises. The standard does this by providing a formal and consistent way to describe and categorise binding mechanisms of various types and strengths. The standard defines three approaches for binding metadata with data objects:

- Encapsulating: The data object together with the metadata is encapsulated within the Binding and is represented by a new composite data object. For example, with eXtensible Markup Language (XML) the use of Binding as a new data object as its root element with the data object and metadata contained directly within the binding element;
- Embedded: The binding information is embedded within the data object and the Binding contains a reference to the information. For example, an XML data object may use a schema that either includes a binding element, or allows it to be extended with arbitrary elements; and

- Detached: The metadata may be stored in a separate structure from the data object with the two linked by reference. However, the binding information and the data objects are always detached. For example, a separate file containing the metadata and the binding element references an XML or JPEG file within a file system via a URI.

6.2.3 NIEM Intelligence Namespace (Intel)

There is no security label metadata indicated in the NIEM 3.0 schema. There appears to be an integration of the NIEM 3.0 Intel namespace and the security metadata associated with the IC ISM, NTK, and TDF namespaces through augmentation, and specified base elements with the specific "anyAttribute" element. Each of the NIEM structures base types (ObjectType, AssociationType, AugmentationType, and MetadataType, as well as SimpleObjectAttributeGroup) incorporates an anyAttributes element as part of its definition. This wildcard allows any attributes to appear that have the IC-ISM namespace, as well as those having the IC-NTK namespace. This is in line with the recommendations of the developer of the IC Trusted Data Framework, and should support ISM, NTK, and the TDF.

6.2.4 Recommendation & Implementation Strategy

The U.S. IC-ISM, IC-TDF and IC-NTK specifications are highly developed and provide comprehensive coverage of the elements, types, and attributes required to provide a level of assurance for electronic information security markings and binding of security metadata to data. The richness of the specifications may be overly complex for the current needs of the investigation into information exchange using NIEM. The adoption and number of departments and agencies within the U.S. actually using these specifications for information exchange is currently unknown.

The NIEM 3.0 Intel namespace makes use of the IC-ISM, IC-TDF and IC-NTK data specifications, by the use of the core anyAttributes element, to provide the electronic information security markings and binding of security metadata to data.
The Bell/Cord3 research and development team is familiar with the NATO STANAG work having participated in a number of NATO STANAG 4774 and 4778 workshops over the last two years. Therefore, the approach taken in the CBSA prototype is to adopt the simpler, and more concise, NATO STANAG 4774 Confidentiality Metadata Label Syntax. This security metadata will be included within the NIEM IEPD N.25 Radiation Device Messaging. This approach allows a single security labeling standard to be used throughout the CBSA prototype, including both within the CBSA domain and for information exchanged with OGDs.

Within DCSS the binding of label security metadata to the data is achieved through the use of symmetrical encryption. In addition, secure hash algorithms and techniques are employed to determine if a security label associated with a file or document has been tampered with. This level of "loose" binding (i.e. not a digital signature) is deemed to provide a sufficient level of assurance within a security domain. However, this approach does not necessarily work well when exchanging information between security domains. Consequently, S/MIME will be used to encrypt the data, and strongly bind the security label to the data, as it transits between security domains.

6.3 Phase 2 Prototype

This section will examine the following aspects of the Phase 2 Prototype:

- DCSS Component Architecture;
- Secure File Retrieval; and
- Secure NIEM Exchange.

6.3.1 DCSS Component Architecture

This section will present a component-level view of the significant elements that will comprise the DCSS implementation in the CBSA prototype. Many of these components have an expanded role from their usage in previous experiments that reflect the new capabilities and maturity that are part of the technology target for this effort. The DCSS components presented in this section, and shown in Figure 15, fall into the categories of:

- Access Management Services;
- Information Protection Services;
- Auditing Services; and
- Support Services.

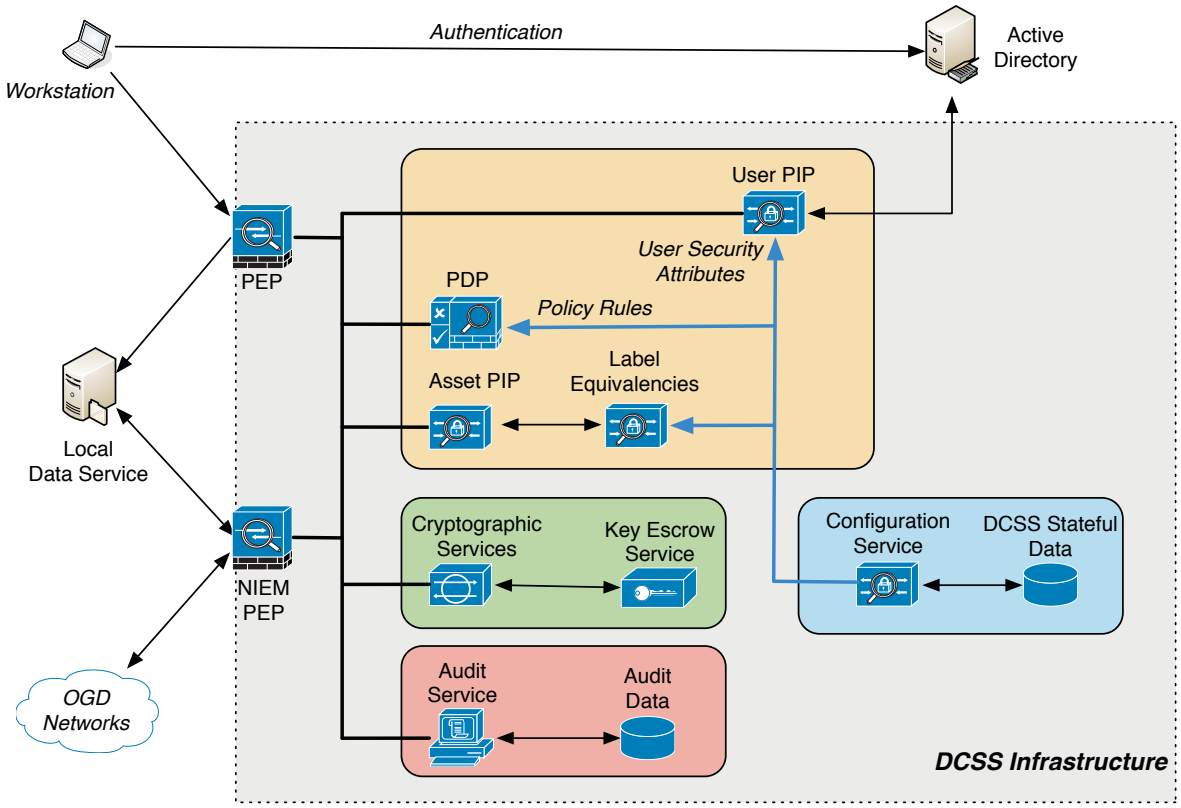

Figure 15 - DCSS Component Architecture

6.3.1.1 Access Management Services

Access Management Services within DCSS include the following:

- Policy Decision Point (PDP) - The PDP will call upon a decision making policy engine to accept the following aspects of a policy decision request:
    - User security attributes as obtained from the userPIP;
    - Asset security attributes as obtained from the assetPIP;
    - Action information as obtained from the PEP (e.g. READ,WRITE);
    - Any environmental attributes that are relevant to the security policy decision;[8]

- User Security Attribute Service (userPIP) - In support of the access management policy structure defined above, the userPIP must, on request, deliver the following security attributes for each DCSS enabled user:
    - Clearance;
    - Nationality;

[8] The CBSA prototype security policy does not use any environmental attribute values as part of the decision making function.

- Group Membership;

- Asset Security Attribute Service (assetPIPs) - the following security attributes must be able to be retrieved from an information asset in order to support access management and policy enforcement operations:
  - Policy Identifier;
  - Classification; and
  - Caveats.

6.3.1.2 Information Protection Services

Information protection services, specifically encryption, retain a similar role as in previous experiments. Assets that are protected by the DCSS infrastructure are encrypted using symmetric key encryption and a key escrow system so that the release of assets is only done when the request is in compliance with policy. Information Protection Services within DCSS include the following:

- Cryptographic Protection Service - The cryptographic services, working in conjunction with a PEP, encrypt information assets when they are to be stored in a secure state within the local network domain and decrypt those assets when they are being release to requesting users or relayed to OGDs. The Cryptographic Protection Service works on files and applies a common algorithm and key length to the cryptographic operation. When a PEP requires encryption on a file asset, the plaintext source and desired cipher text target name of the file is supplied in the call and the protection service will return a key token for the encryption operation. This key token is used to retrieve the actual key that was used in the cryptographic function. When an internal PEP or NIEM PEP requires decryption on a file asset, the cipher text source and desired plaintext target name of the file is supplied in addition to the key token; and

- Key Escrow Service - The key escrow service provides, on demand, a new unique encryption key (of a specified length) and a key token that can be subsequently supplied to retrieve the key at a later time. The key escrow service is a core security service that is called upon by all encryption services for key management functions.

6.3.1.3 Auditing Services

In keeping with DCSS practices, a record of the enforcement of policy decisions is made in a trusted audit store and these events can be used to flag security incidents through linkages with SIEM solutions. Auditing Services within DCSS include the following:

- Trusted Audit Service - Events generated at the national PEPs are sent through the DCA SOA to a trusted audit store. Specifically, these events include the following information:
  - The user that originated the request and the user's security attributes at the time of the request;
  - The requested asset and the security attributes on that asset;
  - The requested action to be performed on the asset;
  - The policy decision that was made by the PDP and an indication of what a request was denied;
  - Meta-information related to the request, including:
    - Time of day the request was made; and
    - The PEP point of origin.

The integrity of the events recorded within the audit store is maintained through the use of cryptographic chain-of-custody block chaining algorithms. Addition, deletion or modification of audit events will be detectable by audit of IT forensic analysts.

Note that not all actions are deemed auditable events. For example, the retrieval of a list of files in a file repository made not be deemed necessary for auditing, whereas the upload and download of files are auditable events.

6.3.1.4 Support Services

In a DCA deployment, the PEPs represent those portions of the solution that ensure that actions on data assets are in accordance with the organizational access management, information protection and auditing strategy. In this way, the PEPs are extended and coordinate access to DCA services to operations that take place in the information technology space: the workstations, data severs and application protocols that link the two.

6.3.2 Secure File Retrieval

The Secure File Retrieval scenario, which is illustrated in Figure 16, consists of the following steps:

- The RSO attempts to retrieve the RDP scan, the CRDS scan and the HCVM scan from the scan gateway;
- The PEP/PDP mediates the access attempt according to policy;
- Assuming that the user is permitted to access the files, the PEP communicates with the DCSS Information Protection Services in order to decrypt the three assets;
- The three labeled assets are downloaded to the RSO's desktop; and
- The RSO creates and labels a fourth data asset (template).

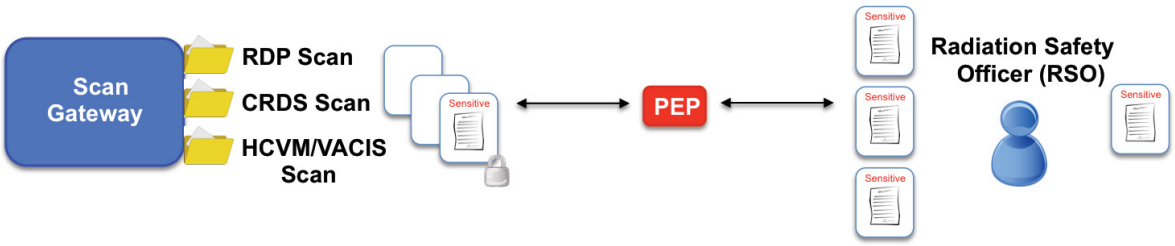

Figure 16 - Secure File Retrieval

6.3.3 Secure NIEM Exchange

The Secure NIEM Exchange has been divided into four pieces in order to facilitate its description. The first piece, which is illustrated in Figure 17, consists of the following steps:

- The RSO creates an email using the Microsoft Outlook client, and labels it using the labeler plugin. The three labeled files from the file server, and the locally generated labeled word file is attached to this email;
- The PEP (email) performs a number of functions. It, in conjunction with the PDP, performs a policy check to ensure that the sender has the right to send the message. It, in conjunction with the Information Protection Services, also encrypts the message, including its attachments; and
- The PEP (email) forwards the protected email on to the MTA.

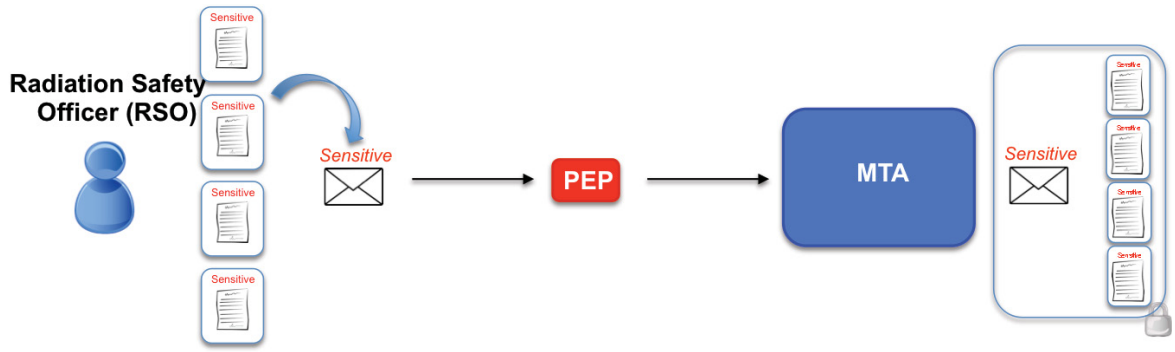

Figure 17 - Secure NIEM Exchange (1 of 4)

The second piece, which is illustrated in Figure 18, consists of the following steps:

- Based on the recipient email address, the MTA send the protected email to the NIEM PEP;
- The NIEM PEP receives the email and performs the following actions:
  - Decrypts the email and attachments;
  - Creates the N.25 document and embeds the four labeled files as base64 encoded elements;
  - Creates an email containing the original message body, the security label, and the N.25 document;
  - Secures the email using S/MIME; and
- The email is forwarded to the appropriate OGD.

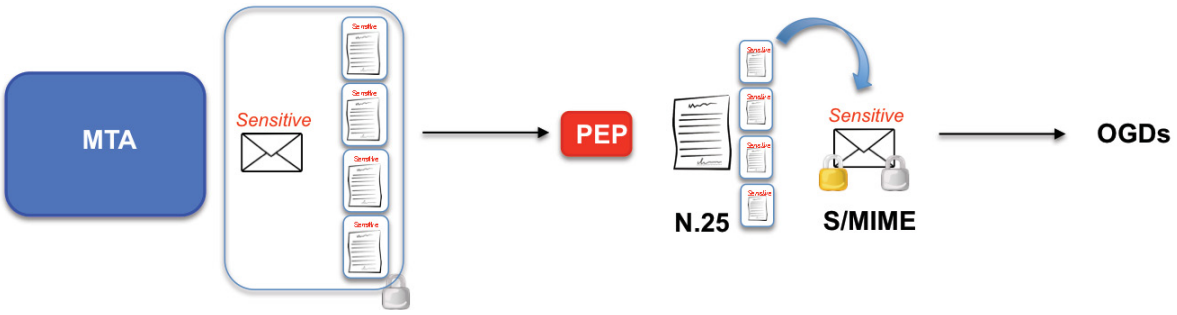

Figure 18 - Secure NIEM Exchange (2 of 4)

The third piece, which is illustrated in Figure 19, consists of the following steps as the email is sent to the CNSC domain:

- The CNSC recipient receives the S/MIME encrypted email;
- The CNSC recipient uses his/her private decryption key to decrypt the email and CBSA's public verification key to verify the digital signature; and

- The CNSC recipient is able to read the email message and open the attachment (N.25 document with four embedded files).

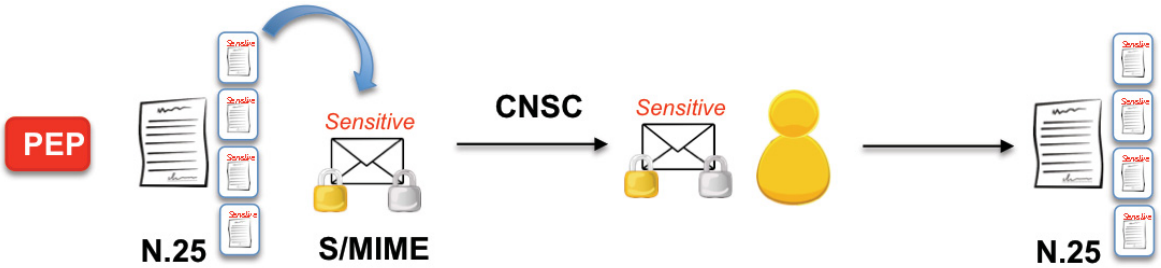

Figure 19 - Secure NIEM Exchange (3 of 4)

The fourth piece, which is illustrated in Figure 20, consists of the following steps as the email is sent to the HC domain:

- The HC NIEM PEP receives the email and performs the following operations:
  - Decrypts the email using the HC private decryption key and verifies the digital signature using the CBSA public verification key;
  - Extracts the four labeled files from the N.25 document;
  - Creates an email containing the original message body, the security label, and the four labeled documents; and
- The HC NIEM PEP sends the email to the HC MTA (not shown), which then forwards it to the HC recipient.

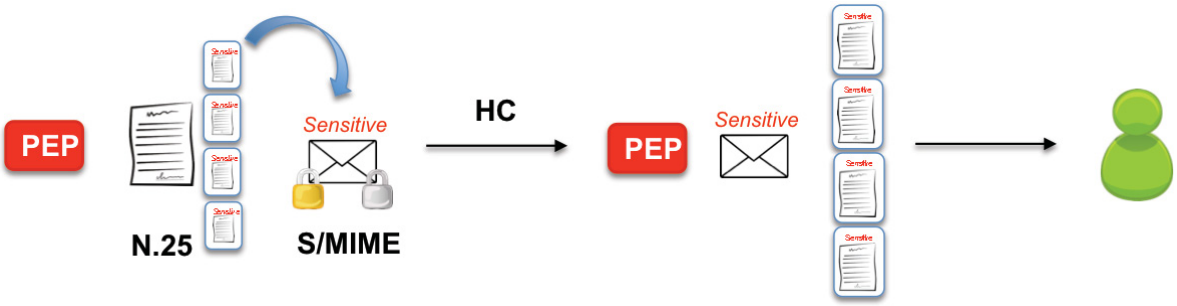

Figure 20 - Secure NIEM Exchange (4 of 4)

6.4 Testing Summary

This section describes the prototype test plan and results. Complete testing information for Phase 2 can be found in Annex C. The testing during Phase 2 shall be in support of a

secure NIEM information exchange between CBSA and OGDs. Specifically, this section will describe the following aspects of Phase 2 testing:

- Testing Overview;
- Types of Tests; and
- Test Results.

6.4.1 Testing Overview

The testing during Phase 2 shall extend the Phase 1 testing through the introduction of the DCSS and the addition of a NIEM PEP in the Health Canada domain, into the prototype and evaluation environment. Phase 1 provided the capability to support a NIEM CBRN information exchange between the CBSA and a Health Canada domain. The principal target of the experiment in Phase 2 is, using CBSA CBRN data, to address the ability of a CBSA RSO user sourcing DCSS protected documents/files from the Scan Gateway file server, and sending an email with file attachments, to HC and the CNSC, using a NIEM PEP to apply the NIEM transformation. There will be two types of receiving domains; HC will have a NIEM transformation and translation capability, while CNSC will only have the ability to receive and interpret NIEM translations (no transformation capability).

6.4.2 Types of Tests

In Phase 2 there will be two types of tests:

- Secure File Retrieval Tests – functional tests to ensure that the RSO is able to retrieve encrypted scan data from the file server; and
- Email Tests – functional tests to ensure that the N.42 data can be sent, as secure (encrypted and digitally signed) NIEM structured data, from the CBSA to OGDs using email as the transport mechanism.

6.4.3 Test Results

As can be seen in Annex C, the Phase 2 prototype successfully passed all twelve testing scenarios.

6.5 Additional IRT Design Detail

The prototype of the IRT application will be implemented in Phase 3 of the CBSA prototype. However, additional detail was developed, and presented to CBSA on 18 December 2015. This additional detail, which is depicted in Figure 21 and Figure 22,

shows how DCSS, will be used to provide students with a subset of the view that RSOs will have. Specifically, a student will be able to create a data record (database server) from a protected N42 asset (stored on the file server) without seeing the content. The student is only entitled to see the records and fields that they have a policy right to see. In contrast, the RSO should be able, subject to policy, to view the full record.

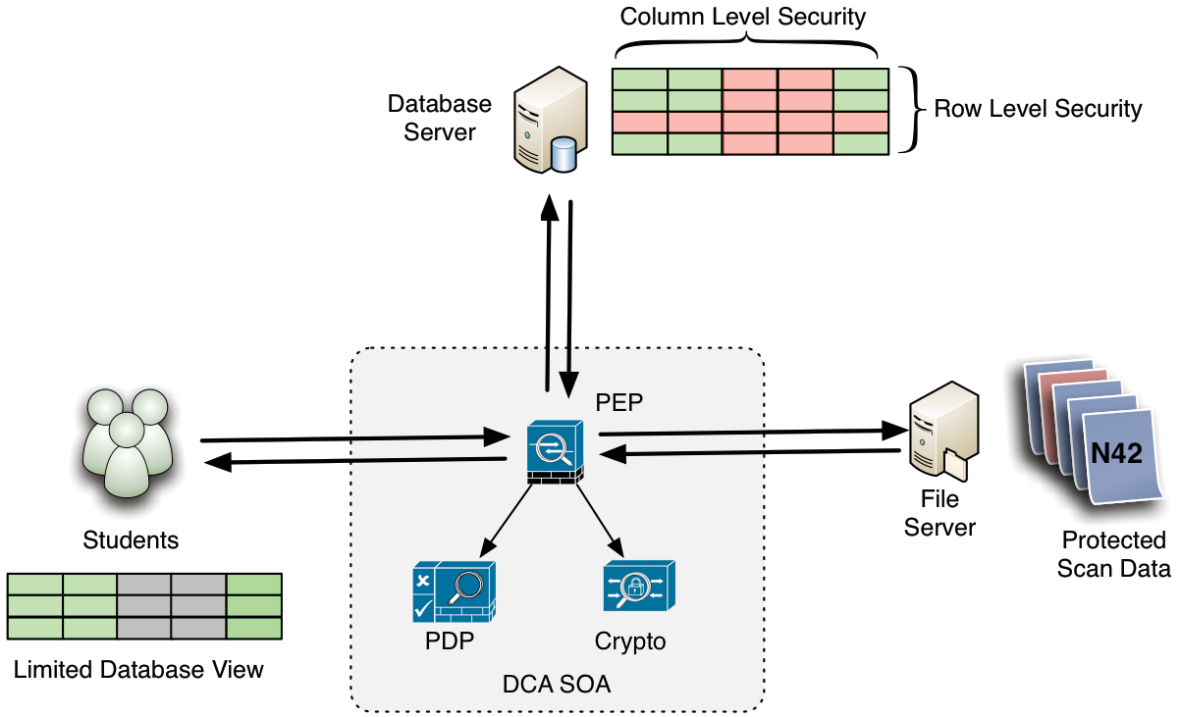

Figure 21 - Student Limited IRT View

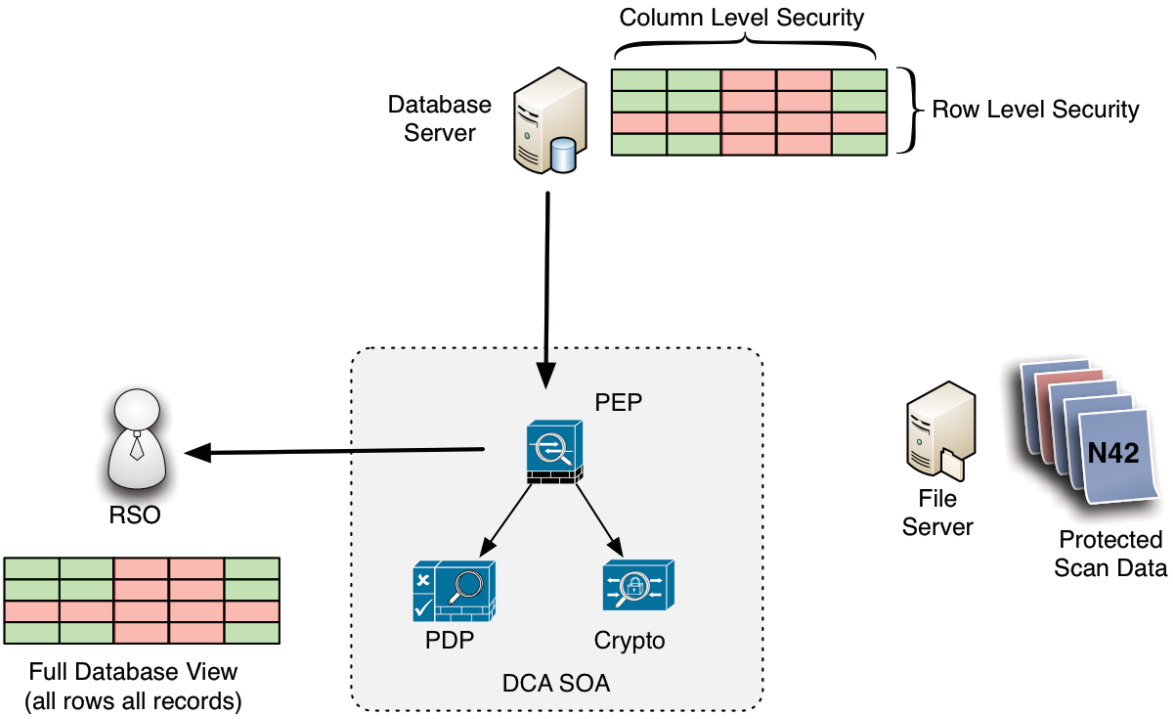

Figure 22 - RSO Full IRT View

7 Phase 3 – Policy Mediated NIEM Exchange

This section of the report will document Phase 3 of the CBSA prototype. Specifically, this section will examine the following:

- Phase 3 Prototype; and
- Testing Summary.

7.1 Phase 3 Prototype

The Phase 3 Prototype has been divided into the following steps, in order to facilitate the explanation:

- Automatic Labelling;
- Secure File Retrieval;
- IRT Prototype;
- Policy Mediated NIEM Exchange;
- CNSC Email Receipt; and
- HC Email Receipt & File Retrieval.

7.1.1 Automatic Labelling

Automatic labelling, which is illustrated in Figure 23, is a new component for Phase 3. In the previous phase, scan data was labelled, encrypted, and stored in a file share on the scan gateway at the beginning of the scenario. In the final iteration of the prototype, scan data, specifically the N.42 documents and image files, gets placed in a directory on the Scan Gateway. This is intended to mimic the current CBSA environment in which scan data is typically uploaded to a gateway. The Labeller, which is constantly polling the directory, detects the new files and performs the following operations:

- Security Labelling – Based on the contents of the file, the Labeller determines the appropriate security label, consisting of both sensitivity and caveat, and creates the appropriate security metadata (Custom.XML);
- Metadata Creation – Based on the contents of the file, including the location of the Scan Gateway, the Labeller creates the appropriate metadata (N.25.XML);
- Container Creation – The Labeller creates a ZIP container in which to store the file, the security metadata, and the metadata; and
- Container Encryption – The Labeller encrypts to the ZIP container so that the file is protected while stored on the Scan Gateway.

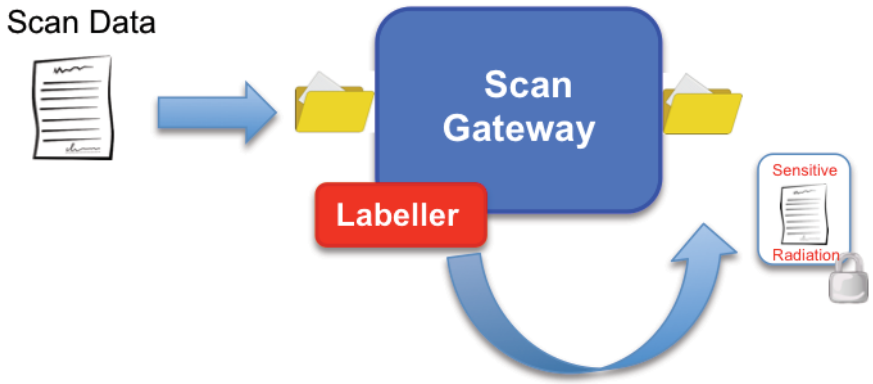

Figure 23 - Automatic Labelling

7.1.2 Secure File Retrieval

The secure file retrieval step, which is illustrated in Figure 24, is similar in most respects to what was demonstrated in Phase 2, with two notable differences; caveats and another type of user (student). In Phase 3, caveats have been added to the data. These caveats include Nuclear, Radiation, Health, and Chemical. Consequently, when the user attempts to retrieve a file from the Scan Gateway, there is a policy check to see if the user is cleared to access the classification of that data, but also whether or not the user is entitled to access the caveat. In the case of the RSO, he is entitled to access the scan data, whereas the student is not.

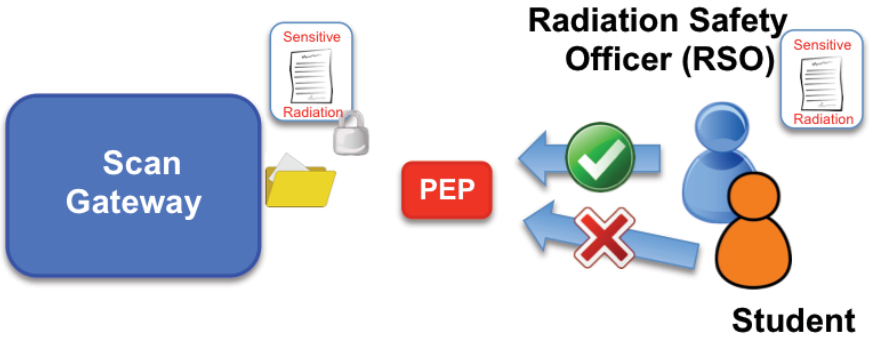

Figure 24 - Secure File Retrieval

7.1.3 IRT Prototype

Phase 3 includes a prototype of the IRT application. In this prototype a student is able to load data into the database, without accessing it directly. Specifically, the student is presented with a list of files stored on the Scan Gateway. From these files, the user chooses which ones to load into the database. Since it is the IRT application that retrieves

and loads the file, the student never accesses the file directly. This is illustrated in Figure 25 and Figure 26. Once the data has been loaded into the database, the student and the RSO will attempt to access the data. As can be seen in Figure 27, Figure 28, and Figure 29, the RSO is provided with a full database view based, while the student only has a limited database view. Access to the database was determined by the same policy check that was used for the secure file retrieval in Section 7.1.2.

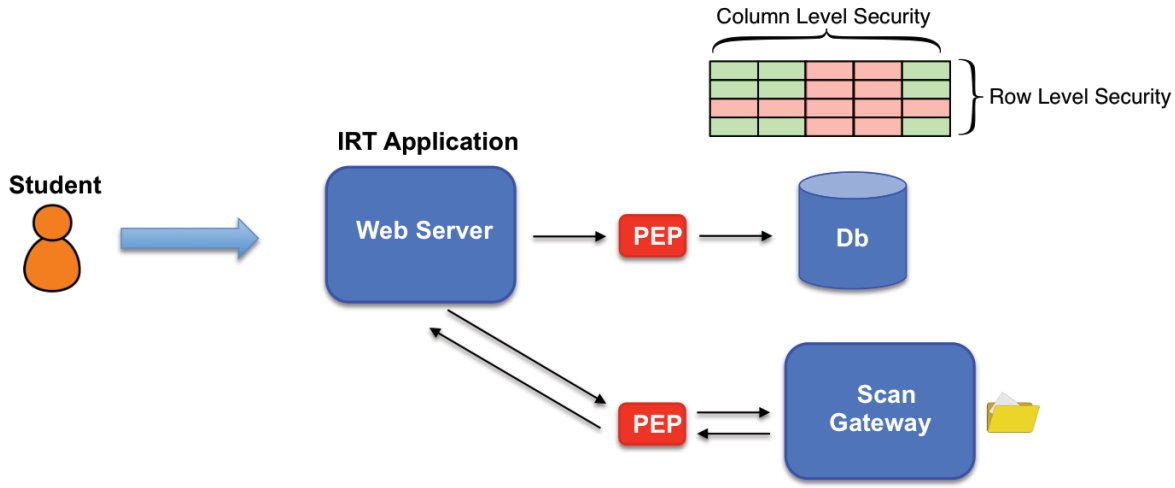

Figure 25 - IRT Prototype: Load Data

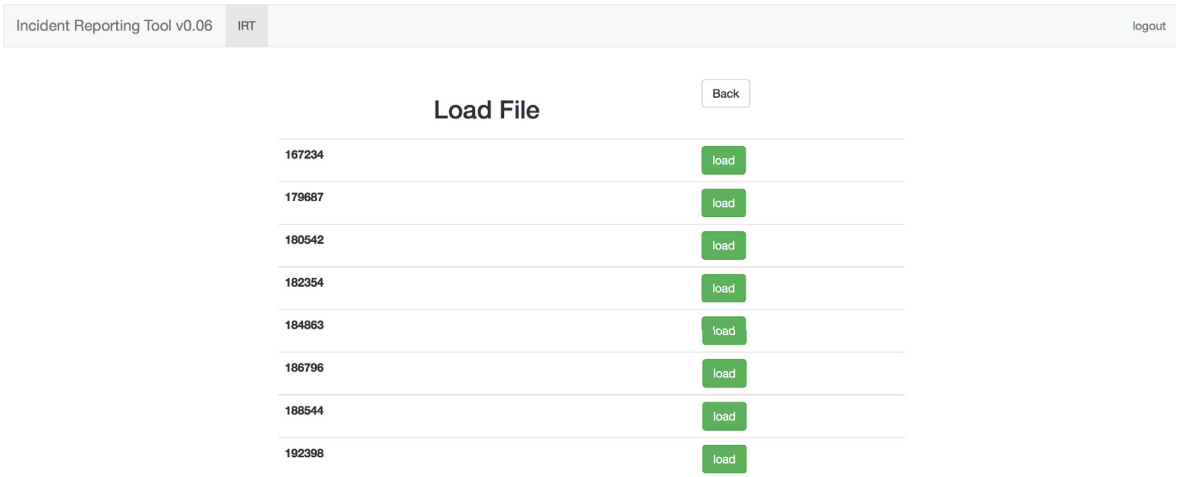

Figure 26 - IRT Prototype: Load Data

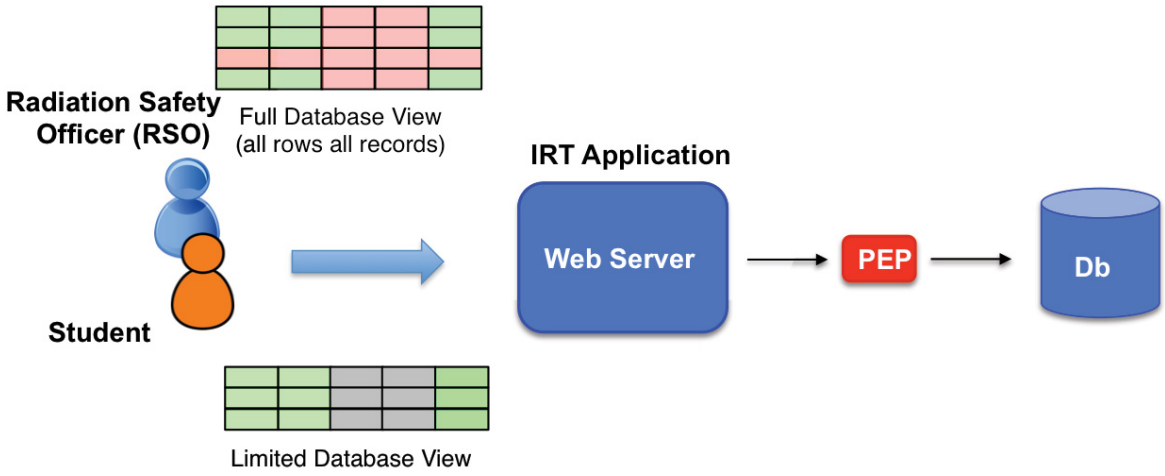

Figure 27 - IRT Prototype: Retrieve Data

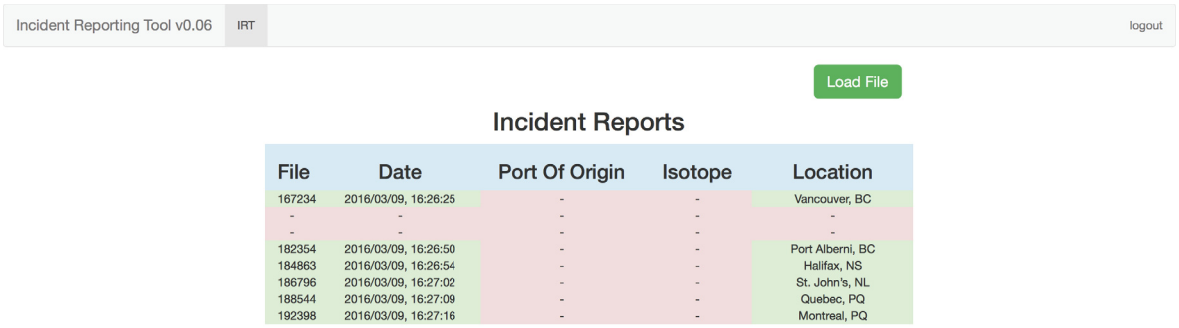

Figure 28 - IRT Prototype: Retrieve Data (Student)

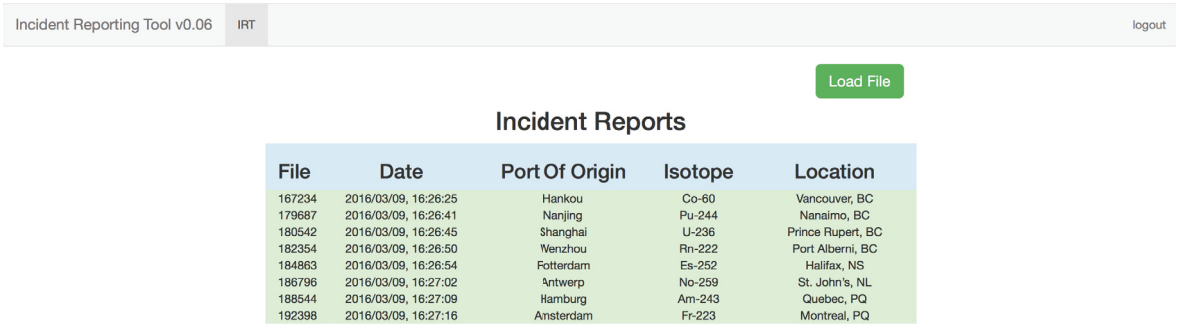

Figure 29 - IRT Prototype: Retrieve Data (RSO)

7.1.4 Policy Mediated NIEM Exchange

One major difference between the Phase 2 prototype and the Phase 3 prototype is policy mediation on release. In other words, a policy check, and specifically a releasability check, is performed on all data leaving the CBSA domain. In Figure 30 the RSO composes his email to the OGDs but accidentally includes a document with the caveat Chemical. When the email gets to the CBSA NIEM PEP, a check is performed on each file in order to determine its releasability. In this particular case, only the caveats Radiation, Nuclear, and Health are releasable to CNSC and HC. Consequently, the releasability check fails, the email is prevented from leaving the CBSA domain, and the RSO is notified of the failure. In Figure 31 the RSO once again composes his email to the OGDs but this time only includes the requisite files. When the email gets to the CBSA NIEM PEP the releasability check is once again made. However, this time it is successful.

One other difference between Phase 2 and Phase 3 is important to note. In Phase 2 the CBSA NIEM PEP created the N.25 document and embedded the four files within. This also occurs in Phase 3, but this time the metadata associated with each file is also included within the N.25 document. As with Phase 2, the N.25 document is included in an email message which is signed using the CBSA private signing key and encrypted using the OGDs' public encryption key.

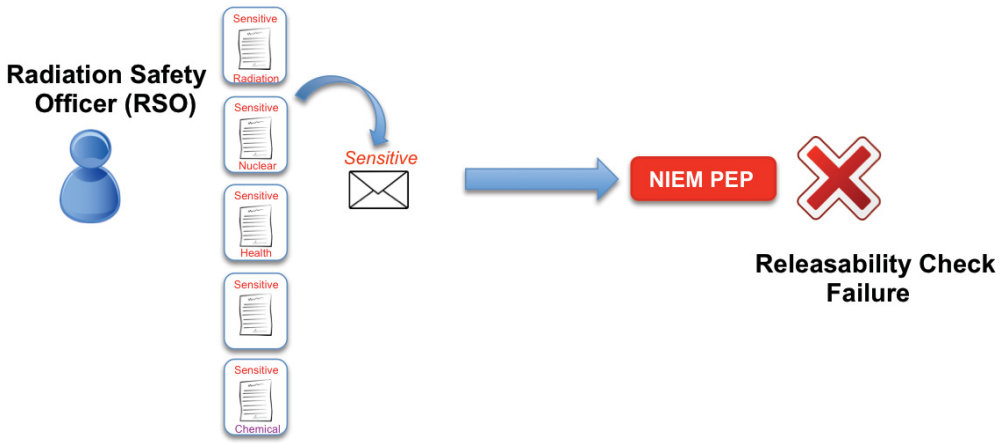

Figure 30 - Policy Mediated NIEM Exchange: Failure

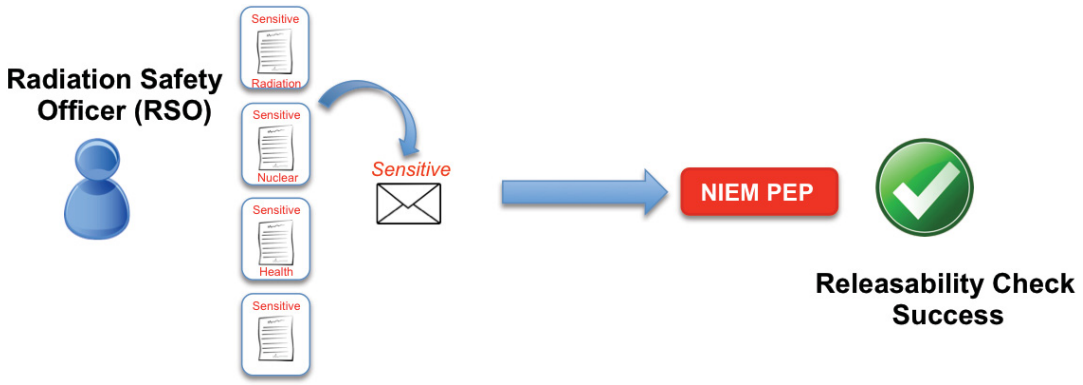

Figure 31 - Policy Mediated NIEM Exchange: Success

7.1.5 CNSC Email Receipt

The CNSC scenario is pretty much the same as Phase 2. Upon receipt of the email, the CNSC Point Of Contact (POC) uses his private decryption key to decrypt the email and CBSA's public verification key to verify the digital signature. The CNSC POC can then view the N.25 document, complete with metadata and embedded files.

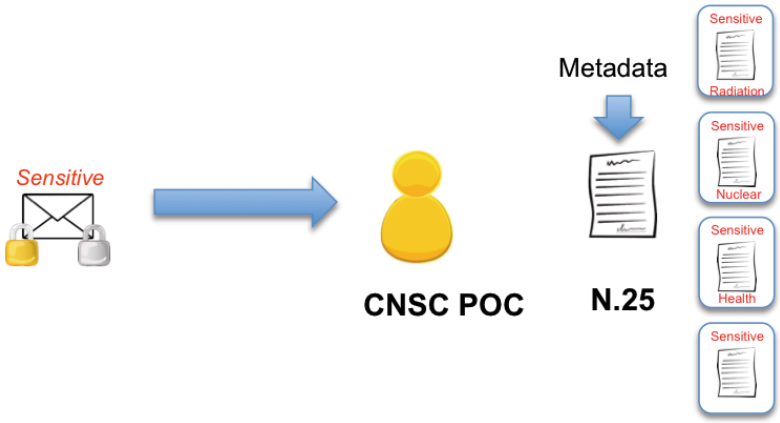

Figure 32 - CNSC Email Receipt

7.1.6 HC Email Receipt & Secure File Retrieval

The most notable changes in this step are the inclusion of three separate HC POCs and a HC file server. When the encrypted and digitally signed email arrives at the HC NIEM PEP, it is decrypted using HC's private decryption key and the digital signature is validated using CBSA's public verification key. Once complete, the embedded files are extracted from the N.25 document and stored securely in the HC file server. An email is created that

includes the body from the original email and links to each of the files stored on the file server. The metadata that was included in the N.25 document for each file is included in the email in order to provide a description of each of the files. The email is sent to each of the HC POCs. This process is illustrated in Figure 33. When each of the HC POCs receives the email they can attempt to access the files by clicking on the links. Access to the files is mediated according to policy. In this scenario each HC POC is permitted to access one of the caveats (Nuclear, Radiation, or Health) but not the others. This is illustrated in Figure 34.

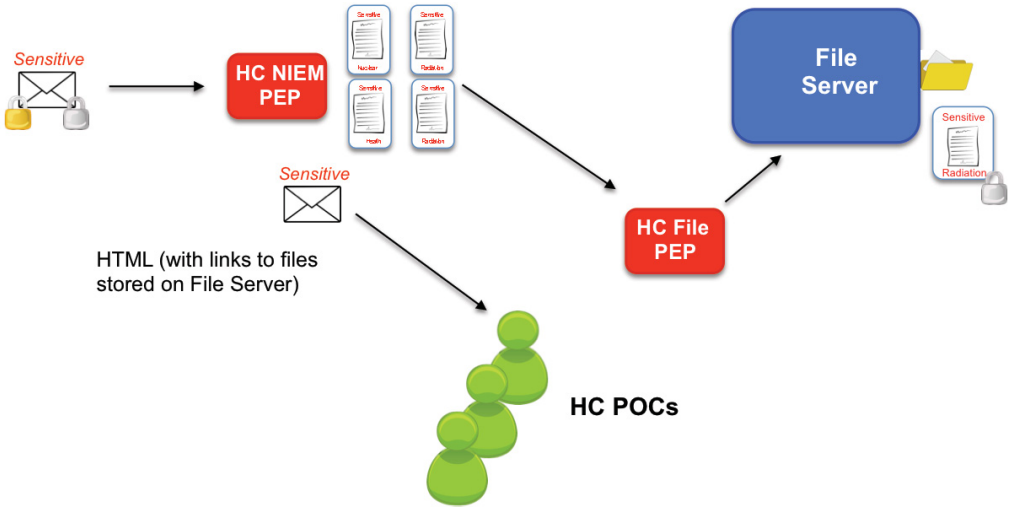

Figure 33 - HC Email Receipt

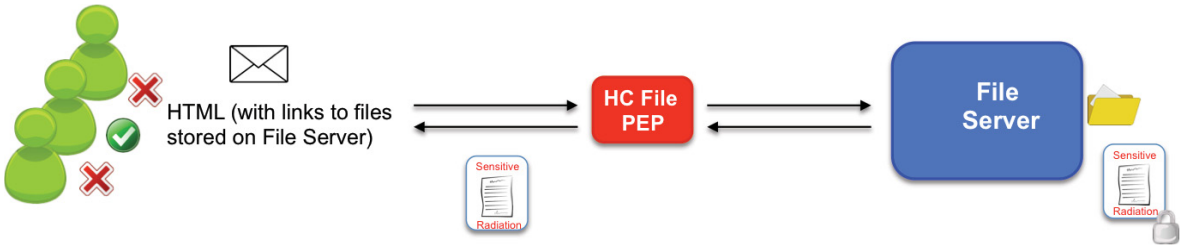

Figure 34 - HC Secure File Retrieval

7.2 Testing Summary

This section summarizes the Phase 3 testing. Complete testing information for Phase 3 can be found in Annex D. The testing during Phase 3 shall be in support of a policy mediated NIEM information exchange between CBSA and OGDs. The Phase 3 testing has provided the evidence that:

a) The file auto security labelling function worked as expected and the security label and event metadata associated with the file could be bound together in a container;
b) The IRT provided the evidence that the DCS system could be used to permit a user with restricted permissions to load files into a protected database and another user with a higher level of permissions, could fully access the incident data previously loaded by a restricted user;
c) The email and file transfer to other government departments testing provided the evidence that:
    - A user in the CBSA domain could have the dissemination of files and email restricted by DCS based on the users Community of Interest (caveat) membership;
    - A user in the CBSA domain could send a number of files to different users in the HC domain and the recipients would only be permitted to access a subset of the files for which they have the requisite privileges;
    - The event metadata associated with each security labelled file could be extracted from the file and shown in the email of the recipient. This provided each recipient of the email with a high level view of the event with the associated N42 formatted file; and
    - An email with an attached N42 file could be sent to another government department, in this case CNSC, which does not have a DCS capability – and the data could still be correctly interpreted in the N42 format.

8 Conclusion & Recommendations

The CBSA Data-Centric Security NIEM Prototype demonstrated how data-centric security can facilitate the secure exchange of standardized information with other organizations subject to policy. Specifically, it achieved the following three project objectives; standardized exchange of radiation scan data with OGDs, securing this exchange, and securing scan data, including access to applications containing scan data, within the CBSA prototype domain environment.

It is important to note that while NIEM provides an information exchange framework with which to facilitate data transformation, it does not provide the complete set of standards required for immediate implementation. Consequently, even with NIEM, data transformation initiatives are not trivial and will necessitate significant involvement from a variety of stakeholders within the organization.

In order to capitalize on the capability demonstrated within this prototype, it is recommended that CBSA pilot this capability within a production environment. Specifically, the intent would be to provide a standardized container with which to transport any type of scan data within the CBSA production environment. The DCSS information protection architecture would be used to secure the container, and the sensitive scan data embedded within, regardless of where it is stored or transits within the production environment.

9 References

[Reference 1] G. Henderson & D. Seguin, *Data Centric Security for CBSA Operations – SAMSON Database Protection for the CBSA SensorNet*, Version 1.01 Draft, Bell Canada/Cord3 Innovation, March 2014;

[Reference 2] A. Magar & G. Henderson, *Data-Centric Security and Information Sharing via NIEM*, Version 1.0, Bell Canada/Cord3 Innovation, March 2015; and

[Reference 3] *Information Interoperability – Architecture Vision*, Version 1.1 (Draft), CBSA, September 2, 2015.

Annex A – Radnet Response Chart

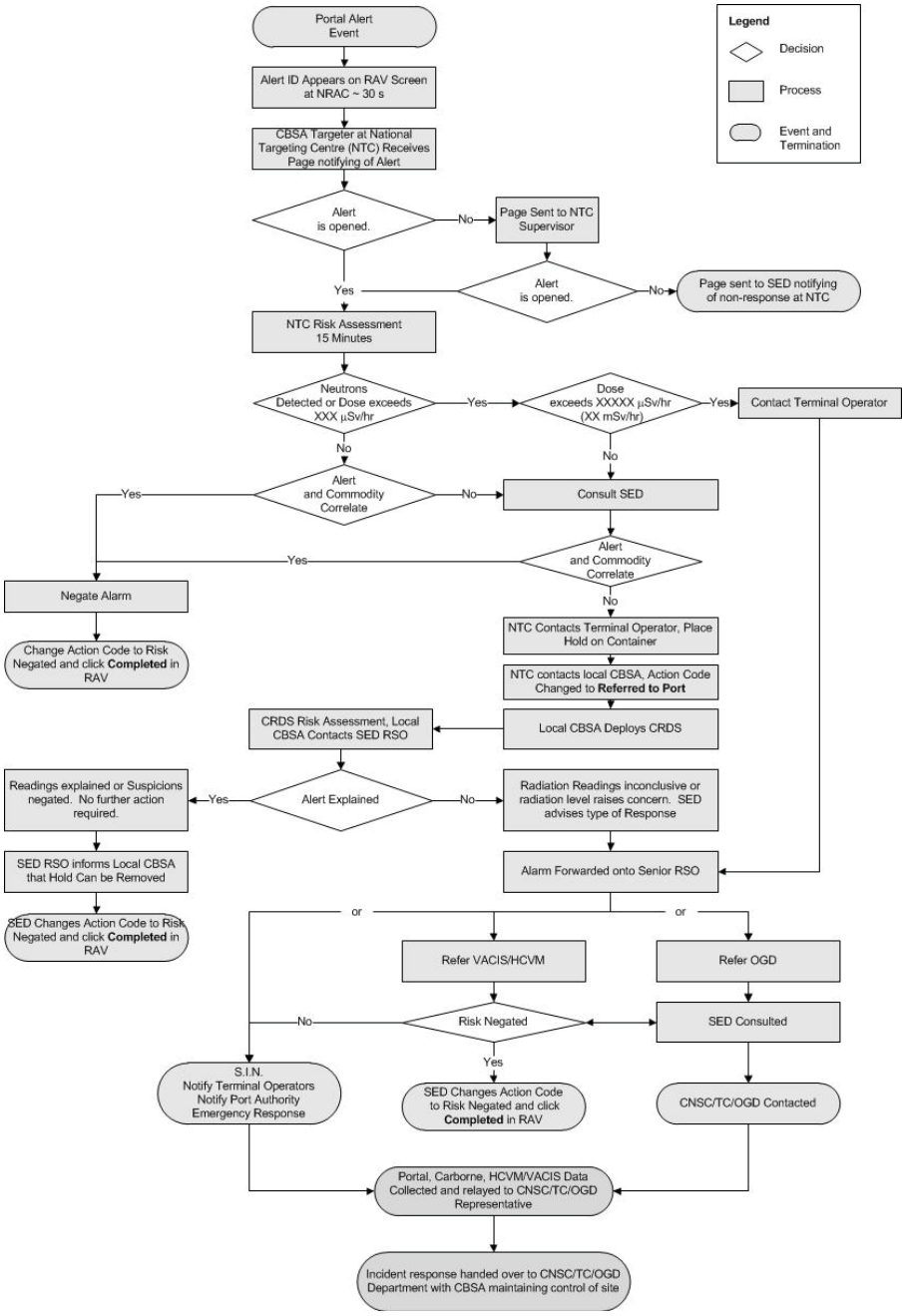

Figure 35 - RADNET Response Chart

Annex B – Phase 1 Test Plan, Procedures & Results

The Phase 1 test plan, procedures and results section is comprised of the following:

- NIEM Test Overview and Conditions;
- Test Scenario Assumptions;
- Test Scenarios; and
- Test Results.

NIEM Test Overview and Conditions

This section provides an overview of the NIEM tests and describes the steps that will be carried out to establish connectivity prior to commencement of the NIEM tests. Specifically, the actual NIEM testing will be performed under the following conditions:

- A series of simple CBSA to OGD connectivity tests will be carried out to establish that the network for the domains is operational and ready for the experiment;
    - A ping test will be carried out by the CBSA to the OGDs to ensure basic connectivity,
    - An email without attachments will be transmitted from the CBSA to the OGDs to ensure that an email can be processed correctly,
    - A simulated OGD without the NIEM PEP will run NIEM conformance tests to ensure it is functioning correctly, and
    - A simulated OGD with a NIEM PEP will run NIEM conformance tests to ensure it is functioning correctly;
- A suite of XML transformational and Email functional tests will be conducted to ensure that the system under test using the NIEM PEP is operational;
- The Project Technical Lead will review all test data and approve the system to enter the CBSA NIEM Testing; and
- Proceed with start of CBSA NIEM Tests.

Test Scenario Assumptions

This section will describe known assumptions including required documents used across each of the tests. Test Scenarios will be generated to provide a high level view of the tests that will be carried out. A number of Test Cases will be developed for each Test Scenario. Test Scenarios for Phase I will be predicated on the following:

- There is no DCS capability;
- Only CBSA will have a NIEM PEP install, the OGDs will not have a NIEM PEP installed; and
- There are no performance or scalability tests planned for Phase 1.

The following documents located in the /home/manager/Documents folder should be copied to the client workstations to be used by the testers:

- CarborneFile.42;
- 184863.12n42; and
- attachment1.docx.

Test Scenarios

This section will provide a description of each of the test scenarios.

Table 7 - Test Scenarios

Test Scenario	Description
Test Scenario 1 Carborne Single Transform	• Overview: This test will verify that a carborne file can be sent by the CBSA RSO as an Outlook email attachment, using the N.42 format, to the outbound NIEM PEP and transmitted in a N.25 format to a recipient in a different OGD domain. • Actions: The RSO within the cbsa.org domain, will send an email message with the carborne file as an attachment, to hcpoc1@hc.com. • Results: Thehcpoc1@hc.com will confirm the arrival of the email with the attachment, open the attachment and inspect the radmessage.n25 to determine that the structure of the attachment is in accordance with NIEM 2.1.
Test Scenario 2 Portal Single Transform	• Overview: This test will verify that a portal scan file can be sent by the CBSA RSO as an Outlook email attachment, using the N.42 format, to the outbound NIEM PEP, and transmitted in a N.25 format to a recipient in a different OGD domain. • Actions: The RSO within the cbsa.org domain, will send an email message with the portal scan file as an attachment, to hcpoc1@hc.com. • Results: The hcpoc1@hc.com will confirm the arrival of the email with the attachment, open the attachment and inspect the radmessage.n25 to determine that the structure of the attachment is in accordance with NIEM 2.1.
Test Scenario 3 HCVM Single Transform	• Overview: This test will verify that a HCVM scan image file can be sent by the RSO as an Outlook email attachment, using the N.42 format, to the outbound NIEM PEP, and transmitted in a N.25 format to a recipient, in a different OGD domain. • Actions: The RSO within the cbsa.org domain, will send an email message with the HCVM image file as an attachment, to hcpoc1@hc.com. • Results: The hcpoc1@hc.com will confirm the arrival of the email with the attachment, open the attachment and inspect the radmessage.n25 to determine that the structure of the attachment is in accordance with NIEM 2.1.

Test Scenario	Description
Test Scenario 4 Word Document Single Transform	• Overview: This test will verify that a Microsoft Word document file can be sent by the CBSA RSO as an Outlook email attachment, using the N.42 format, to the outbound NIEM PEP, and transmitted in a N.25 format to a recipient in a different OGD domain. • Actions: The RSO within the cbsa.org domain, will send an email message with the Word document file as an attachment, to hcpoc1@hc.com. • Results: The hcpoc1@hc.com will confirm the arrival of the email with the attachment, open the attachment and inspect the radmessage.n25 to determine that the structure of the attachment is in accordance with NIEM 2.1.
Test Scenario 5 Incorrectly Structure N.42 File	• Overview: This test will verify that a carborne file, which is a malformed N.42 format, but still sent by the CBSA RSO as an Outlook email attachment, using the attached malformed N.42 format file. • Actions: The RSO within the cbsa.org domain, will send an email message with the malformed carborne file as an attachment, to hcpoc1@hc.com. • Results: The hcpoc1@hc.com will confirm the arrival of the email with the attachment, open the attachment and inspect the radmessage.n25 to determine that the structure of the attachment is in accordance with NIEM 2.1 but the base64 encoding is different from the correctly formed carborne.n42 file used in Test 1.

Test Results

This section provides a detailed description of the results for each test scenario.

Table 8 – Test 1 Scenario Results

Test Scenario 1 Carborne Single Transform	Input attached file: CarborneFile.n42	Received attached file: radmessage.n25	Pass/Fail
	Actions	Results	
XML Validation by inspection	The RSO within the cbsa.org domain, will send an email message with the carborne file as an attachment, to hcpoc1@hc.com.	The hcpoc1@hc.com domain will confirm the arrival of the email with the attachment, open the attachment and inspect the “radmessage.N25” to determine that the structure of the attachment is in accordance with NIEM 2.1 outlined in Table1	Pass

Test Scenario 1 Carborne Single Transform	**Input attached file: CarborneFile.n42**	**Received attached file: radmessage.n25**	**Pass/Fail**
	Actions	**Results**	
Proof of transport integrity The base64 encoding will be inspected for consistency between input and received message.	Incoming.msg Base64 encoding: PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz4NCjxONDJJbnN0cnVtZW50RGF0YSB4bWxucz0iaHR0cDovL3BoeXNpY3MubmlzdC5nb3YvRGl2aXNpb25zL0Rpdjg0Ni9HcDQvQU5T……Mm86U3BhblRlbXBsYXRlPg0KICA8L3JzaW40Mm86UnNpTWVhc3VyZW1lbnQ+DQo8L040Mkluc3RydW1lbnREYXRhPg0K	Radmessage.N25 Base64 encoding: PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz4NCjxONDJJbnN0cnVtZW50RGF0YSB4bWxucz0iaHR0cDovL3BoeXNpY3MubmlzdC5nb3YvRGl2aXNpb25zL0Rpdjg0Ni9HcDQvQU5T……..Mm86U3BhblRlbXBsYXRlPg0KICA8L3JzaW40Mm86UnNpTWVhc3VyZW1lbnQ+DQo8L040Mkluc3RydW1lbnREYXRhPg0K	Pass
Transform test This is a one-time test to provide evidence that the Carborne n42 file can be received as a base64 blob and converted back using a conversion tool to the Carborne n42 XML format.	Open the radmessage.n25 file. Carry out a copy and paste of the contents of the tag <cbrn:BinaryBase64Object> </cbrn:BinaryBase64Object> Paste the base64 blob into a file called carborneBase64.txt. Use the base64toXMLConverter.py tool to convert the base64 blob back to a file called carborneBase64.xml and inspect this file to ensure it is identical to the input Carborne File.n42	Test carried out using the "base64toXMLConverter" python tool. The received file radmessage BinaryBase64Object generates the carborneBase64.xml which is consistent with the input file Carborne File.n42	Pass

Table 9 – Test 2 Scenario Results

Test Scenario 2 Portal Single Transform	**Input attached file: 184863.12n42**	**Received attached file: radmessage.n25**	**Pass/Fail**
	Actions	**Results**	

Test Scenario 2 Portal Single Transform	Input attached file: 184863.12n42	Received attached file: radmessage.n25	Pass/Fail
	Actions	**Results**	
XML Validation by inspection	The RSO within the cbsa.org domain, will send an email message with the portal scan 184863.12n42 file as an attachment, to hcpoc1@hc.com.	The hcpoc1@hc.com domain will confirm the arrival of the email with the attachment, open the attachment and inspect the "radmessage.N25" to determine that the structure of the attachment is in accordance with NIEM 2.1 outlined in Table1	Pass
Proof of transport integrity The base64 encoding will be inspected for consistency between input and received message.	Incoming.msg Base64 encoding: PD94bWwgdmVyc2lvbj0iMS4wIj8 +DQo8UmFkSW5zdHJ1bWVudE RhdGEgeG1sbnM9Imh0dHA6Ly9 w aHlzaWNzLm5pc3QuZ292L040Mi 8yMDExL040MiIgeG1sbnM6eHN pPSJodHRwOi8vd3d3LnczLm9y ………….dXJhdGlvbj4NCiAgICA gICAgICAgIDxDb3VudERhdGE+ MDwvQ291bnREYXRhPg0KICAg ICAgICA8L0dyb3NzQ291bnRzPg 0KICAgIA==	Radmessage.N25 Base64 encoding: PD94bWwgdmVyc2lvbj0iMS4wIj8+ DQo8UmFkSW5zdHJ1bWVudERh dGEgeG1sbnM9Imh0dHA6Ly9waH lzaWNzLm5pc3QuZ292L040Mi8yM DExL040MiIgeG1sbnM6eHNpPSJo dHRwOi8vd3d3LnczLm9y………… .dXJhdGlvbj4NCiAgICAgICAgICAgI DxDb3VudERhdGE+MDwvQ291bn REYXRhPg0KICAgICAgICA8L0dyb 3NzQ291bnRzPg0KICAgIA==	Pass

Table 10 – Test 3 Scenario Results

Test Scenario 3 HCVM Single Transform	**Input attached file: HCVM.jpg**	**Received attached file: radmessage.n25**	**Pass/Fail**
	Actions	**Results**	
XML Validation by inspection	The RSO within the cbsa.org domain, will send an email message with the HCVM image file as an attachment, to hcpoc1@hc.com	The hcpoc1@hc.com domain will confirm the arrival of the email with the attachment, open the attachment and inspect the “radmessage.N25” to determine that the structure of the attachment is in accordance with NIEM 2.1 outlined in Table1	Pass
Proof of transport integrity The base64 encoding will be inspected for consistency between input and received message.	Incoming.msg Base64 encoding: /9j/4AAQSkZJRgABAQEAYABgAAD/2wBDAAIBAQIBAQICAgICAgICAwUDAwMDAwYEBAMFBwYHBwcG………….sQj8x5vD8/74YIJ36iMk4jO9nCr0Unawr9Pq4XSfHmp3n7S2veGWa3/sfTvDWnanEgTEwnnur6N2LdCpW3QAdQUb1FAH5/8A/Bv9rOofEL9on9qDx0+h6noui+PNSs9csI7uIKYGutc8T3j2zun7p54IuoRJ5RZAXX53bc7fp1RRQB//2Q==	Radmessage.N25 Base64 encoding: /9j/4AAQSkZJRgABAQEAYABgAAD/2wBDAAIBAQIBAQICAgICAgICAwUDAwMDAwYEBAMFBwYHBwcG………….sQj8x5vD8/74YIJ36iMk4jO9nCr0Unawr9Pq4XSfHmp3n7S2veGWa3/sfTvDWnanEgTEwnnur6N2LdCpW3QAdQUb1FAH5/8A/Bv9rOofEL9on9qDx0+h6noui+PNSs9csI7uIKYGutc8T3j2zun7p54IuoRJ5RZAXX53bc7fp1RRQB//2Q==	Pass

Table 11 – Test 4 Scenario Results

Test Scenario 4 Word Document Single Transform	**Input attached file:** **184863.12n42**	**Received attached file:** **radmessage.n25**	**Pass/Fail**
	Actions	**Results**	
XML Validation by inspection	The RSO within the cbsa.org domain, will send an email message with a Microsoft Word document as an attachment, to hcpoc1@hc.com.	The hcpoc1@hc.com domain will confirm the arrival of the email with the attachment, open the attachment and inspect the “radmessage.N25” to determine that the structure of the attachment is in accordance with NIEM 2.1 outlined in Table1	Pass

Test Scenario 4 Word Document Single Transform	Input attached file: 184863.12n42	Received attached file: radmessage.n25	Pass/Fail
	Actions	**Results**	
Proof of transport integrity The base64 encoding will be inspected for consistency between input and received message.	Incoming.msg Base64 encoding: UEsDBBQABgAIAAAAIQCshlBXj gEAAMAFAAATAAgCW0NvbnRl bnRfVHlwZXNdLnhtbCCiBAIooA ACAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAA…………..BgAI AAAAIQALThPj7wEAAPEDAAAQ AAAAAAAAAAAAAAAAAMWkAA Bkb2NQcm9wcy9hcHAueG1sUE sFBgAAAAANAA0ATAMAAOqnA AAAAA==	Radmessage.N25 Base64 encoding: UEsDBBQABgAIAAAAIQCshlBXjg EAAMAFAAATAAgCW0NvbnRlbn RfVHlwZXNdLnhtbCCiBAIooAACA AAAAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAAAAAAAAAAAAA AAAAAAAAAAAAAAAAAAAAAAA AAAAAA…………..BgAIAAAAIQAL ThPj7wEAAPEDAAAQAAAAAAAA AAAAAAAAAMWkAABkb2NQcm9 wcy9hcHAueG1sUEsFBgAAAAAN AA0ATAMAAOqnAAAAAA==	Pass

Table 12 – Test 5 Scenario Results

Test Scenario 5 Incorrectly Structure N.42 File	Input attached file: Malformed_CarborneFile.n42	Received attached file: radmessage.n25	Pass/Fail
	Actions	**Results**	
XML Validation by inspection	The RSO within the cbsa.org domain, will send an email message with a malformed xml file, Malformed_CarborneFile.n42, as an attachment, to hcpoc1@hc.com.	The hcpoc1@hc.com domain will confirm the arrival of the email with the attachment, open the attachment and inspect the “radmessage.N25” to determine that the structure of the attachment is in accordance with NIEM 2.1 outlined in Table1	Pass
Proof of transport integrity The base64 encoding will be inspected for consistency between input and received message.	Incoming.msg Base64 encoding: PD94bWwgdmVyc2lvbj0iMS4wIiB lbmNvZGluZz0iVVRGLTgiPz4NCj xONDJJbnN0cnVtZW50RGF0YS B4bWxucz0iaHR0cDovL3BoeXN pY3MubmlzdC5nb3YvRGl2aXNp b25zL0Rpdjg0Ni9HcDQvQU5T… ………..IDwvcnNpbjQybzpTcGFu VGVtcGxhdGU+DQogIDwvcnNpb jQybzpSc2lNZWFzdXJlbWVudD4 NCjwvTjQySW5zdHJ1bWVudER hdGE+DQo=	Radmessage.N25 Base64 encoding: PD94bWwgdmVyc2lvbj0iMS4wIiBlb mNvZGluZz0iVVRGLTgiPz4NCjxO NDJJbnN0cnVtZW50RGF0YSB4b Wxucz0iaHR0cDovL3BoeXNpY3M ubmlzdC5nb3YvRGl2aXNpb25zL0 Rpdjg0Ni9HcDQvQU5T…………..I DwvcnNpbjQybzpTcGFuVGVtcGxh dGU+DQogIDwvcnNpbjQybzpSc2l NZWFzdXJlbWVudD4NCjwvTjQyS W5zdHJ1bWVudERhdGE+DQo=	Pass

Test Scenario 5 Incorrectly Structure N.42 File	**Input attached file:** **Malformed_CarborneFile.n42**	**Received attached file:** **radmessage.n25**	**Pass/Fail**
	Actions	**Results**	
Proof of Error The base64 encoding will be inspected for inconsistency between Test1 input and received message	Test 1 Correctly formed Carborne.N42 base64 encoding: PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz4NCjxONDJJbnN0cnVtZW50RGF0YSB4bWxucz0iaHR0cDovL3BoeXNpY3MubmlzdC5nb3YvRGl2aXNpb25zL0Rpdjg0Ni9HcDQvQU5T……Mm86U3BhblRlbXBsYXRlPg0KICA8L3JzaW40Mm86UnNpTWVhc3VyZW1lbnQ+DQo8L040Mkluc3RydW1lbnREEYXRhPg0K	Test 1 radmessage.N25 Base64 encoding: PD94bWwgdmVyc2lvbj0iMS4wIiBlbmNvZGluZz0iVVRGLTgiPz4NCjxONDJJbnN0cnVtZW50RGF0YSB4bWxucz0iaHR0cDovL3BoeXNpY3MubmlzdC5nb3YvRGl2aXNpb25zL0Rpdjg0Ni9HcDQvQU5T……..Mm86U3BhblRlbXBsYXRlPg0KICA8L3JzaW40Mm86UnNpTWVhc3VyZW1lbnQ+DQo8L040Mkluc3RydW1lbnREEYXRhPg0K	Pass

Annex C – Phase 2 Test Plan, Procedures & Results

The testing during Phase II shall extend the Phase 1 testing through the introduction of the DCSS and the addition of a NIEM PEP in the HC domain, into the prototype and evaluation environment. Phase I provided the capability to support a NIEM CBRN information exchange between the CBSA and a Health Canada domain. The principal target of the experiment in Phase II is, using CBSA CBRN data, to address the ability of a CBSA RSO user sourcing DCSS protected documents/files from the Scan Gateway file server, and sending an email with file attachments, to HC and the CNSC, using a NIEM PEP to apply the NIEM transformation. There will be two types of receiving domains; HC will have a NIEM transformation and translation capability, while CNSC will only have the ability to receive and interpret NIEM translations (no transformation capability).

Background

The tests are based on work carried out during previous phases of the SAMSON project that was sponsored by Defence Research and Development Canada (The Secure File Service and email experiments are based on the following conditions:

- Does the current user have the policy right to create a security label for an object?
- Does the current user have the policy right to transfer an object given the object's current security label?
- Does the user of file services have the policy right to see the directory listing of this file?
- Does the user of file services have the policy right to retrieve the file from the file services?
- Sender Policy ID: Will be a fixed value, such as CBSA 1.2.3.4;
- Classification: Fixed value of "SENSITIVE";
- COI: The Community of Interest or Caveats to whom the file can be sent to and received by will be CHEM, NUCLEAR, and RAD (although selectable using the labeller tool these attributes will not be used in this Phase II testing); and
- Releasability: Other Government Departments to whom the file can be shared with for example //HC/CNSC (Releasability is based on the recipient email address in Phase II testing).

Test Users

The table below provides a listing of each user and their DCS security attributes:

Table 13 - Test Users

Test Users		POLICY		
ID	PWD	COI	CLASSIFICATION	ACTION
CBSARSO1	Pass4Cord3		SENSITIVE	R/W
HCPOC1	Pass4Cord3		SENSITIVE	R/W
CNSCPOC1	Pass4Cord3		SENSITIVE	R/W

CBSA Secure File Retrieval Test Scenarios

FS File Scenario 1(User creates file security label and retrieve file)
Overview: This scenario will demonstrate the CBSARSO1 creating a file with the security label information and posting it to a file server. The same user will retrieve the file. The file security label data will be selected and permitted by the DCS Policy for the user as read and write.

Actions: The CBSARSO1 user will append the security label information of PrivacyMark = CLEAR; PolicyIdentifier=NATO; Classification=Sensitive; TagName=Additional Sensitivity; Type=Restrictive; GenericValue=RSO; to the Carborne.n42 file and save it as Carborne.n25 in the local drive, move the file to the /data DCS protected folder mounted on the workstation. The CBSARSO1 user will retrieve the security labeled Carborne.n25 file from the DCS protected data file folder and open the document, using a file name of Carborne.zip, and ensure that the item1.xml security label information is correct.

FS File Scenario 2 (NoCryptoBypass)
Overview: This scenario will demonstrate a DCS user copying a DCS protected file by using winscp to a local machine and the user will attempt to gain access to the file contents.

Actions: A DCS user uses winscp to copy a DCS protected file to a local machine folder (bypassing the filePEP)and the user will attempt to open the file.

FS File Scenario 3 (Unlabelled File)
Overview: This scenario will demonstrate a DCS user opening a file and not assigning a security label to the file and attempting to save it to the DCS protected file server.

Actions: A DCS user will open a Word document and not assign a security label to it, then attempt to save the document to a Protected File server.

NIEM Email Translation Test Scenarios

This testing will verify that:

a) The CBSA transmitting NIEM PEP can correctly transform the local N.42 formatted files, an image file, and a Microsoft Word document, and be transported as N.25 formatted files to OGDs;
b) The HC receiving NIEM PEP will transform the NIEM N.25 formatted files back to the N.42 format; and
c) The CNSC receives an email in the NIEM N.25 format.

NIEM Scenario 1 Carborne double transform to HC

Overview: This test will verify that a Carborne.n42 file can be sent by the CBSA RSO as an Outlook email attachment, using the N.42 format, to the outbound NIEM PEP and transmitted in a N.25 format to a NIEM PEP in the HC environment to the HC recipient.

Actions: The RSO within the cbsa.com domain, will send an email message with the carborne file as an attachment, to hcpoc1@hc.com.

Results: Thehcpoc1@hc.com will confirm the arrival of the email with the attachment, open the attachment and inspect the attachment to determine that the structure of the attachment is in the original document format.

NIEM Scenario 2Carborne single transform to CNSC

Overview: This test will verify that a Carborne.n42 file can be sent by the CBSA RSO as an Outlook email attachment, using the N.42 format, to the outbound NIEM PEP and transmitted in a N.25 format to the CNSC environment and to the CNSC recipient.

Actions: The RSO within the cbsa.com domain, will send an email message with the carborne file as an attachment, to cnscpoc1@cnsc.com.

Results: The cnscpoc1@cnsc.com will confirm the arrival of the email with the attachment, open the attachment and inspect the attachment to determine that the structure of the attachment is in the N.25 format.

NIEM Scenario 3Portal double transform to HC

Overview: This test will verify that a Portal.n42 file can be sent by the CBSA RSO as an Outlook email attachment, using the N.42 format, to the outbound NIEM PEP and transmitted in a N.25 format to a NIEM PEP in the HC environment to the HC recipient.

Actions: The RSO within the cbsa.com domain, will send an email message with the portal file as an attachment, to hcpoc1@hc.com.

Results: Thehcpoc1@hc.com will confirm the arrival of the email with the attachment, open the attachment and inspect the attachment to determine that the structure of the attachment is in the original document format.

NIEM Scenario 4 Portal single transform to CNSC
Overview: This test will verify that a portal scan file can be sent by the CBSA RSO as an Outlook email attachment, using the N.42 format, to the outbound NIEM PEP, and transmitted in a N.25 format to a recipient in a different OGD domain.

Actions: The CBSARSO1 within the cbsa.com domain, will send an email message with the portal scan file (184863.12n42) as an attachment, to cnscpoc1@cnsc.com.

Results: The cnscpoc1@cnsc.com will confirm the arrival of the email with the attachment, open the attachment and inspect the structure of the attachment is in the N.25 format.

NIEM Scenario 5 HCVM double transform to HC
Overview: This test will verify that a HCVM image file can be sent by the CBSA RSO as an Outlook email attachment, using the N.42 format, to the outbound NIEM PEP and transmitted in a N.25 format to a NIEM PEP in the HC environment to the HC recipient.

Actions: The RSO within the cbsa.com domain, will send an email message with the HVCM image file an attachment, to hcpoc1@hc.com.

Results: Thehcpoc1@hc.com will confirm the arrival of the email with the attachment, open the attachment and open the attachment to determine that the structure of the attachment is in the original document format.

NIEM Scenario 6 HCVM single transform to CNSC
Overview: This test will verify that a HCVM scan image file can be sent by the RSO as an Outlook email attachment, using the N.42 format, to the outbound NIEM PEP, and transmitted in a N.25 format to a recipient, in a different OGD domain.

Actions: The RSO within the cbsa.org domain, will send an email message with the HCVM image file as an attachment, to cnscpoc1@cnsc.com.

Results: The cnscpoc1 will confirm the arrival of the email with the attachment, open the attachment and inspect the file to determine that the structure of the attachment is in accordance with the N.25 format.

NIEM Scenario 7 Word Document double transform to HC
Overview: This test will verify that a Word document (docx) file can be sent by the CBSA RSO as an Outlook email attachment, using the N.42 format, to the outbound NIEM PEP and transmitted in a N.25 format to a NIEM PEP in the HC environment, and to the HC recipient.

Actions: The RSO within the cbsa.com domain, will send an email message with the word document as an attachment, to hcpoc1@hc.com.

Results: Thehcpoc1@hc.com will confirm the arrival of the email with the attachment, open the attachment and open the word document attachment to determine that the structure of the attachment is in the original document format.

NIEM Scenario 8 Word Document single transform
Overview: This test will verify that a Microsoft Word document file can be sent by the CBSA RSO as an Outlook email attachment, using the N.42 format, to the outbound NIEM PEP, and transmitted in a N.25 format to a recipient in a different OGD domain.

Actions: The RSO within the cbsa.com domain, will send an email message with the Word document file as an attachment, to cnscpoc1@cnsc.com.

Results: The cnscpoc1@hc.com will confirm the arrival of the email with the attachment, open the attachment and inspect the structure of the attachment is in accordance with the N.25 format.

NIEM Scenario 9 Incorrectly structure N.42 file
Overview: This test will verify that a carborne file, which is a malformed N.42 format, but still sent by the CBSA RSO as an Outlook email attachment, using the attached malformed N.42 format file.

Actions: The CBSARSO1 within the cbsa.com domain, will send an email message with the malformed carborne file as an attachment, to hcpoc1@hc.com.

Results: The hcpoc1@hc.com will confirm the arrival of the email with the attachment, open the attachment and inspect the structure of the attachment is in accordance with the N.25 format but the base64 encoding is different from the correctly formed carborne.n42 file used in Test 1

The N.25 Transformation

The N.42 XML to N.25 XML is based on the following transform shown in Table 14.

Table 14 - N.42 XML to N.25 XML

```
<?xml version="1.0" encoding="UTF-8"?>
<n25rd:RadiationDeviceMessage xmlns:i="http://niem.gov/niem/appinfo/2.0"
 xmlns:niem-xsd="http://niem.gov/niem/proxy/xsd/2.0"
 xmlns:unece="http://niem.gov/niem/unece_rec20-misc/2.0"
 xmlns:fips_10-4="http://niem.gov/niem/fips_10-4/2.0"
 xmlns:s="http://niem.gov/niem/structures/2.0"
 xmlns:nga="http://niem.gov/niem/nga/2.0"
 xmlns:nc="http://niem.gov/niem/niem-core/2.0"
 xmlns:iso_3166="http://niem.gov/niem/iso_3166/2.0"
 xmlns:cbrn="http://www.dhs.gov/niem/dndo/CBRNSchema"
 xmlns:scr="http://niem.gov/niem/domains/screening/2.0"
 xmlns:it="http://niem.gov/niem/domains/internationalTrade/2.0"
 xmlns:cbrncl="http://www.dhs.gov/niem/cbrndomain"
 xmlns:fbi="http://niem.gov/niem/fbi/2.0"
 xmlns:intel="http://niem.gov/niem/domains/intelligence/2.0"
 xmlns:n25rd="http://www.dhs.gov/niem/dndo/CBRNSchema/N25"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://www.dhs.gov/niem/dndo/CBRNSchema/N25
file:/C:/Users/Brent/Documents/Cord3/CBSA/N25/IEPD%20N.25/Version%201/Version%201.1.40/Schemas/Exch
ange/radiationdevicemessage-x.xsd">
  <n25rd:MessageContent>
    <n25rd:EventDataFiles>
      <cbrn:DetectionEventUUID>DetectionEventUUID2</cbrn:DetectionEventUUID>
                          <n25rd:EventDataFile>
        <cbrn:Binary>
          <cbrn:RemarkText>RemarkText24</cbrn:RemarkText>
          <cbrn:BinaryMetadata>
            <cbrn:VersionIdentifier>VersionIdentifier0</cbrn:VersionIdentifier>
            <cbrn:DeclassManualReviewIndicator>false</cbrn:DeclassManualReviewIndicator>
            <cbrn:VersionEffectiveDateTime>2006-05-04T18:13:51.0</cbrn:VersionEffectiveDateTime>
            <cbrn:SecurityClassificationCode>CONFIDENTIAL</cbrn:SecurityClassificationCode>
            <cbrn:DocumentPrivacyActIndicator>false</cbrn:DocumentPrivacyActIndicator>
            <cbrn:ClassificationReasonText>ClassificationReasonText0</cbrn:ClassificationReasonText>
            <cbrn:ClassifiedByText>ClassifiedByText0</cbrn:ClassifiedByText>
            <cbrn:DeclassDate>2006-05-04</cbrn:DeclassDate>
            <cbrn:DeclassEventText>DeclassEventText0</cbrn:DeclassEventText>
            <cbrn:DeclassExceptionText>DeclassExceptionText0</cbrn:DeclassExceptionText>
          </cbrn:BinaryMetadata>
          <cbrn:DetectionEventUUID>DetectionEventUUID3</cbrn:DetectionEventUUID>
          <cbrn:BinaryFileUUID>BinaryFileUUID0</cbrn:BinaryFileUUID>
          <cbrn:BinaryFileName>$FileName</cbrn:BinaryFileName>
          <cbrn:BinaryCaptureData>
            <cbrn:BinaryCaptureStartTime>2006-05-04T18:13:51.0</cbrn:BinaryCaptureStartTime>
            <cbrn:BinaryCaptureDuration>P1Y2M3DT1H10M0S</cbrn:BinaryCaptureDuration>
```

```
            <cbrn:BinaryCaptureDeviceID>BinaryCaptureDeviceID0</cbrn:BinaryCaptureDeviceID>
            <cbrn:BinaryCaptureDeviceCategoryCode>ASP</cbrn:BinaryCaptureDeviceCategoryCode>
            <cbrn:MIMEEncodingCode>7bit</cbrn:MIMEEncodingCode>
            <cbrn:MIMEContentCode>323</cbrn:MIMEContentCode>
          </cbrn:BinaryCaptureData>

<cbrn:BinaryCompressionDescriptionText>BinaryCompressionDescriptionText1</cbrn:BinaryCompressionDescrip
tionText>
          <cbrn:BinarySequence>
            <cbrn:BinarySequenceIdentifier>BinarySequenceIdentifier0</cbrn:BinarySequenceIdentifier>

<cbrn:BinarySequenceDescriptionText>BinarySequenceDescriptionText0</cbrn:BinarySequenceDescriptionText>
            <cbrn:ImagePerspectiveCode>Bottom</cbrn:ImagePerspectiveCode>
          </cbrn:BinarySequence>
          <cbrn:BinarySubjectCode>Container Image</cbrn:BinarySubjectCode>
          <cbrn:BinaryBase64Object>$FileData</cbrn:BinaryBase64Object>
        </cbrn:Binary>
      </n25rd:EventDataFile>
                        </n25rd:EventDataFiles>
  </n25rd:MessageContent>
</n25rd:RadiationDeviceMessage>
```

Test Procedures

File Service Tests

Table 15 - File Service Tests

Test No.	Test Actions	Expected Results	P/F
FS Test 1	The CBSARSO1 user will append the security label information of PrivacyMark = CLEAR; PolicyIdentifier=NATO; Classification=Sensitive; TagName=Additional Sensitivity; Type=Restrictive; GenericValue=RSO; to the Carborne File.n42 file. Save it as Carborne.n25 in the local drive, move the file to the /data DCS protected folder mounted on the workstation.	item1.xml is: <?xml version="1.0" ?> - <slab:ConfidentialityLabel xmlns:slab="**http://cord3.ca/schema**"> - <slab:ConfidentialityInformation> <slab:PrivacyMark>**CLEAR**</slab:PrivacyMark> <slab:PolicyIdentifier>**NATO**</slab:PolicyIdentifier> <slab:Classification>**Sensitive**</slab:Classification> - <slab:Category TagName="**Additional Sensitivity**" Type="**Restrictive**"> <slab:GenericValue>**RSO**</slab:GenericValue> </slab:Category> <slab:CreationDateTime>**2016-01-25T20:29:45Z**</slab:CreationDateTime> </slab:ConfidentialityInformation> </slab:ConfidentialityLabel>	Pass

	Open the Carborne.n25 file in the \data DCS protected folder	The Carborne.n25 file in the \data DCS protected folder when opened will display the contents of a compressed zip file. The filePEP log file will indicate the Policy Decision of "permit" 1453867672 INFO filePEP : {"result":{"decision":"permit"},"status":{"status code":"0","statusmessage":"OK"}} 1453867673 INFO filePEP : {"subject":{"identity":"CBSARSO1"},"resource":{"source":"/usr/local/apache2/htdocs/data/Carborne.n42", "target":"/usr/local/apache2/aesir/getfile-26305"},"action":{"action":"read"}} 1453867673 DEBUG filePEP : Starting the file PEP unprotect call 1453867673 DEBUG filePEP : Message contains the correct required attributes 1453867673 DEBUG filePEP : The user attribute values have been expanded 1453867673 DEBUG filePEP : The asset attribute values have been expanded 1453867673 INFO filePEP : The policy decision was: permit 1453867673 DEBUG filePEP : Completed the file PEP unprotect file call 1453867673 INFO filePEP : Call completed successfully 1453867673 INFO filePEP : {"result":{"decision":"permit"},"status":{"status code":"0","statusmessage":"OK"}}	
	The CBSARSO1 user will copy the security labeled Carborne.n25 file from the DCS protected data file folder to the c:\data folder on the local drive.	File Carborne.n25 copied from the \data DCS protected folder to the local c:\data folder.	Pass
	Rename the c:\data\Carborne.n25 file to Carborne.zip. Open the document, and ensure that the item1.xml security label information is unchanged and correct.	Item1.xml file contents: <?xml version="1.0" ?> <slab:ConfidentialityLabel xmlns:slab="**http://cord3.ca/schema**"> <slab:ConfidentialityInformation> <slab:PrivacyMark>**CLEAR**</slab:PrivacyMark> <slab:PolicyIdentifier>**NATO**</slab:PolicyIdentifier>	Pass

		<slab:Classification>**Sensitive**</slab:Classification> <slab:Category TagName="**Additional Sensitivity**" Type="**Restrictive**"> <slab:GenericValue>**RSO**</slab:GenericValue> </slab:Category> <slab:CreationDateTime>**2016-01-25T20:29:45Z**</slab:CreationDateTime> </slab:ConfidentialityInformation> </slab:ConfidentialityLabel>	
	item1.xml when opened in a browser displays the security label bound to the file.	?xml version="1.0" ?> slab:ConfidentialityLabel xmlns:slab="**http://cord3.ca/schema**"> slab:ConfidentialityInformation> slab:PrivacyMark>**CLEAR**</slab:PrivacyMark> slab:PolicyIdentifier>**NATO**</slab:PolicyIdentifier> slab:Classification>**Sensitive**</slab:Classification> slab:Category TagName="**Additional Sensitivity**" Type="**Restrictive**"> slab:GenericValue>**RSO**</slab:GenericValue> </slab:Category> slab:CreationDateTime>**2016-01-25T20:29:45Z**</slab:CreationDateTime> </slab:ConfidentialityInformation> </slab:ConfidentialityLabel>	Pass
File Test 2	Open a new word document file as CBSARSO1	Word file opened with the Cord3 labeller visible in the top rt hand corner of the band.	Pass
	Enter text using =lorem(5)	Text generated	Pass
	Assign security label data of Sensitive and Chemical	Security label generated	Pass
	Save the word document as test4.docx in the mounted data folder	File saved in the data folder	Pass
	Double click on the test4.docx to open the word document	Document opens	Pass
	Click on Security Labeler and confirm the security label is Sensitive and Chemical	Confirmed	Pass
	Using winscp copy the test4.docx file to the c:\test folder	File copied from the filePEP /usr/local/apache2/htdocs/data folder to the local c:\test folder	Pass
	Attempt to open the test4.docx file in the local c:\test folder	Warning displayed that the file test4.docx cannot be opened because there are problems with the contents	Pass
File Test 3	Open a Word document and insert text using =lorem(6)	Text generated	Pass
	Attempt to save the word document as test5.docx to the	Word displays a “File permission error” and the /var/log/aesir/filePEP log indicates	Pass

	mounted DCS protected data folder	1453778360 DEBUG filePEP : Starting the file PEP protect call 1453778360 DEBUG filePEP : Message contains the correct required attributes 1453778360 DEBUG filePEP : The user attribute values have been expanded 1453778360 ERROR filePEP : The call to the asset attribute service returned an error. 1453778360 ERROR filePEP : read_file_from_archive: file <customXml/item1.xml> could not be found in zip archive. 1453778360 ERROR filePEP : Could not interpret the asset attribute response. 1453778360 ERROR filePEP : The call to the asset attribute service returned an error. 1453778360 INFO filePEP : The call to the asset attribute service returned an error. 1453778360 INFO filePEP : Call completed successfully 1453778360 INFO filePEP : {"status":{"statuscode":"1102","statusmessage":"The call to the asset attribute service failed."}}	

NIEM Email Tests

Table 16 - NIEM Email Tests

Test	Actions	Results	P/F
NIEM Test 1	The CBSARSO1 within the cbsa.com domain, will send an email message with a security label of “sensitive” and a “sensitive” security labelled Carborne File.n42 file as an attachment, to hcpoc1@hc.com.	Thehcpoc1@hc.com will confirm the arrival of the email with the Carborne File.n42 attachment. Save the attachment on the local drive in the c:\test folder. Rename the file from Carborne File.n42 to Carborne File.zip. Open Carborne File.zip. The Carborne File.n42 should be displayed. Open the Carborne File.n42 with Wordpad and inspect the document to determine that the structure of the attachment is in the original Carborne.n42 format. Open the displayed folder customXml. The file item1.xml should be displayed.	Pass
	item1.xml displays the file security label in Internet Explorer.	Open the item1.xml file in Internet Explorer and the following should be displayed: ?xml version="1.0" encoding="utf-8" ?> slab:ConfidentialityLabel xmlns:slab="**http://cord3.ca/schema**"> slab:ConfidentialityInformation> slab:PolicyIdentifier>**OrganizationX**</slab:PolicyIdentifier> slab:Classification>**Sensitive**</slab:Classification> slab:Category TagName="**Additional Sensitivity**" Type="**Restrictive**"> GenericValue>**RSO**</GenericValue> </slab:Category> </slab:ConfidentialityInformation> </slab:ConfidentialityLabel>	Pass

NIEM Test 2	The CBSARSO1 within the cbsa.com domain, will send an email message with the carborne.n42 file as an attachment, to cnscpoc1@cnsc.com.	The cnscpoc1@cnsc.com will confirm the arrival of the email with the radmessage.n25 attachment. Open the attachment with Wordpad and inspect the attachment to determine that the structure of the attachment is in the N.25 format as shown in Table 1.	Pass
NIEM Test 3	The CBSARSO1 within the cbsa.com domain, will send an email message with a security label of “sensitive” and a “sensitive” security labelled 184863.12n42 file as an attachment, to hcpoc1@hc.com	Thehcpoc1@hc.com will confirm the arrival of the email with the attachment, Save the attachment on the local drive in the c:\test folder. Rename the file from 184863.12n42 to 184863.zip Open 184863.zip. The 184863.12n42 should be displayed. Open the 184863.12n42 with Wordpad and inspect the document to determine that the structure of the attachment is in the original 184863.12n42 format. Open the displayed folder customXml. The file item1.xml should be displayed.	Pass
	item1.xml displays the file security label in Internet Explorer.	<?xml version="1.0" encoding="utf-8" ?> slab:ConfidentialityLabel xmlns:slab="**http://cord3.ca/schema**"> slab:ConfidentialityInformation> slab:PolicyIdentifier>**OrganizationX**</slab:PolicyIdentifier> slab:Classification>**Sensitive**</slab:Classification> slab:Category TagName="**Additional Sensitivity**" Type="**Restrictive**"> GenericValue>**RSO**</GenericValue> </slab:Category> </slab:ConfidentialityInformation>	Pass

		</slab:ConfidentialityLabel>	
NIEM Test 4	The CBSARSO1 within the cbsa.com domain, will send an email message with the portal scan file (184863.12n42) as an attachment, to cnscpoc1@cnsc.com.	The cnscpoc1@cnsc.com will confirm the arrival of the email with the radmessage.n25 attachment. Open the radmessage.n25 attachment and inspect the structure of the attachment is in the N.25 format as shown in Table 1.	Pass
NIEM Test 5	The CBSARSO1 within the cbsa.com domain, will send an email message with the HVCM image.jpg file an attachment, to hcpoc1@hc.com.	Thehcpoc1@hc.com will confirm the arrival of the email with the HCVM.jpg attachment. Save the attachment on the local drive in the c:\test folder. Rename the file from HCVM.jpg to HCVM.zip Open HCVM.zip. The HCVM.jpg file should be displayed. Open the HCVM.jpg file and the picture should be displayed. Open the displayed folder customXml. The file item1.xml should be displayed.	Pass
	item1.xml displays the file security label in Internet Explorer.	?xml version="1.0" encoding="utf-8" ?> slab:ConfidentialityLabel xmlns:slab="**http://cord3.ca/schema**"> slab:ConfidentialityInformation> slab:PolicyIdentifier>**OrganizationX**</slab:PolicyIdentifier> slab:Classification>**Sensitive**</slab:Classification> slab:Category TagName="**Additional Sensitivity**" Type="**Restrictive**"> GenericValue>**RSO**</GenericValue> </slab:Category> </slab:ConfidentialityInformation> </slab:ConfidentialityLabel>	Pass

NIEM Test 6	The CBSARSO1 within the cbsa.org domain, will send an email message with the HCVM image.jpg file as an attachment, to cnscpoc1@cnsc.com.	The cnscpoc1 will confirm the arrival of the email with the radmessage.n25 attachment, open the attachment and inspect the file to determine that the structure of the attachment is in accordance with the N.25 format as shown in Table 1.	Pass
NIEM Test 7			
Validation by test and inspection	The CBSARSO1 within the cbsa domain, will send an email message with the test3.docx as an attachment, to hcpoc1@hc.com.	The hcpoc1@hc.com domain will confirm the arrival of the email with the attachment, open the attachment using Word and inspect the document footer for the correct sensitivity information.	Pass
NIEM Test 8 Validation by test and inspection	The CBSARSO1 within the cbsa domain will send an email with test3.docx as an attachment to the CNSCPOC1 user in the cnsc domain.	The cnscpoc1@cnsc.com will confirm the arrival of the email with the attachment radmessage.n25. The attachment will be opened by Wordpad and inspect it to conform with the n25 schema shown in Table 1.	
NIEM Test 9 Validation by test and inspection	The CBSARSO1 within the cbsa.com domain, will send an email message with the malformed carborne file as an attachment, to hcpoc1@hc.com.	The hcpoc1@hc.com will confirm the arrival of the email with the MalformedCarborne.n42 attachment, Save the MalformedCarborne.n42 attachment on the local drive in the c:\test folder. Rename the file from MalformedCarborne.n42 to MalformedCarborne.zip Open MalformedCarborne.zip. The Malformed_Carborne.n42 should be displayed. Open the Malformed_Carborne.n42 with Wordpad and inspect the document to determine that the structure of the attachment is in the original Carborne.n42 format, except that the following modification has been made: < ! – –	Pass

		xsi:schemaLocation="http://physics.nist.gov/Divisions/Div846/Gp4/ANSIN4242/2005/ANSIN4242 42 http://physics.nist.gov/Divisions/Div846/Gp4/ANSIN4242/2005/ANSIN4242.xsd" --> Open the displayed folder customXml. The file item1.xml should be displayed.	
	item1.xml displays the file security label in Internet Explorer.	?xml version="1.0" encoding="utf-8" ?> slab:ConfidentialityLabel xmlns:slab="**http://cord3.ca/schema**"> slab:ConfidentialityInformation> slab:PolicyIdentifier>**OrganizationX**</slab:PolicyIdentifier> slab:Classification>**Sensitive**</slab:Classification> slab:Category TagName="**Additional Sensitivity**" Type="**Restrictive**"> GenericValue>**RSO**</GenericValue> </slab:Category> </slab:ConfidentialityInformation> </slab:ConfidentialityLabel>	Pass

Annex D – Phase 3 Test Plan, Procedures & Results

The testing during Phase III shall extend the testing of Phase II with the introduction of the full DCSS into the Health Canada prototype and evaluation environment and the addition of the IRT application, which has been added to the CBSA domain. An automatic security labelling capability will be introduced, whereby N.42 xml files, and Security Incident files will be assigned a security label depending on the files contents. Phase I provided the capability to support a NIEM CBRN Information Exchange between the CBSA and a Health Canada domain. The principal target of the experiment in Phase II was, using CBSA CBRN data, address the ability of a CBSA RSO user sourcing DCS protected documents/files from the Scan Gateway file server, and sending an email with file attachments, to HC and the CNSC, using a NIEM PEP to apply the NIEM transformation. In Phase III the DCS component have been added to the Health Canada domain. There are two types of receiving OGDs; HC will have a NIEM transformation and translation capability, and a full DCSS access control capability. The CNSC will only have the ability to receive and interpret NIEM translations (no re-transformation capability). The IRT database has been added in the CBSA domain.

Test Conditions

The diagram below provides a high level view of the test environment setup:

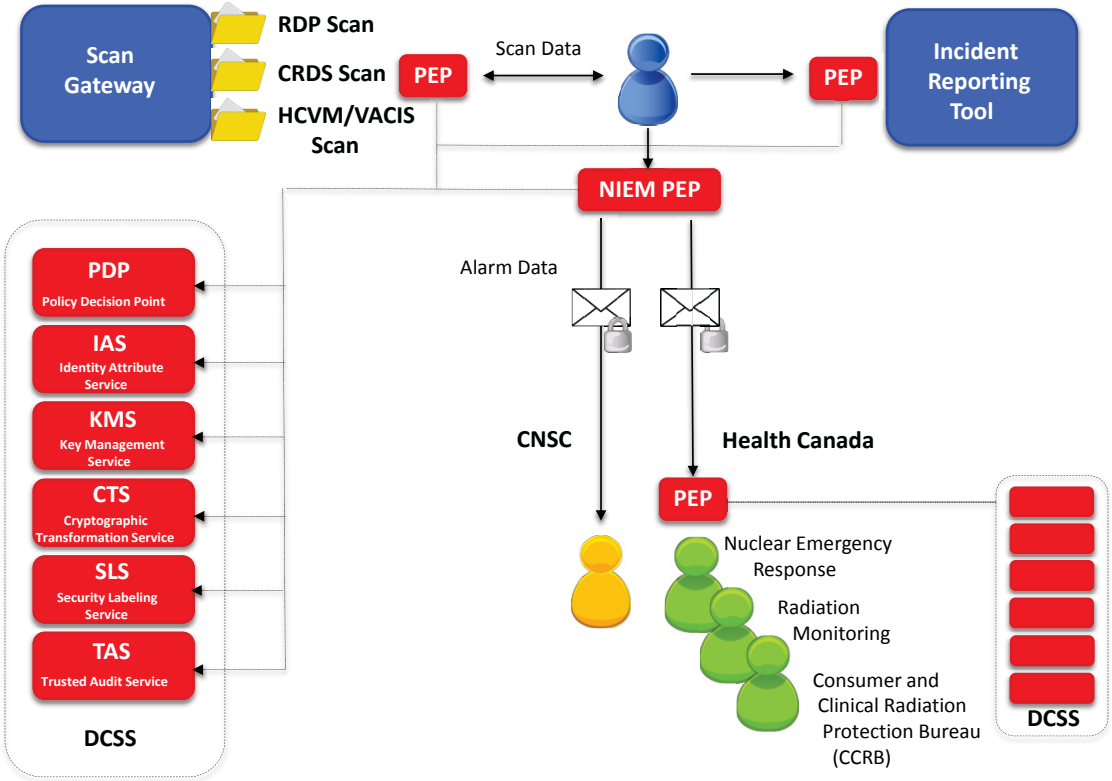

Figure 36 - Phase 3 Test Environment

The file services and email experiments are based on work carried out during previous phases of the CBSA NIEM project. The Secure File Service, IRT, and email experiments are based on the following conditions:

- Does the current user have the policy right to create a security label for a file or email?
- Does the current user have the policy right to transfer a file, load a file, or send an email given the file or email's current security label?
- Does the user of file services have the policy right to see the directory listing of this file?
- Does the user of file services have the policy right to retrieve the file from the file services?
- Classification: Fixed value of "SENSITIVE"; or "NONE"
- COI: The Community of Interest or Caveats to be used will be: NUCLEAR, HEALTH, CHEMICAL and RADIATION
- Releasability: Will be based on a COI.

Detailed Test Environment

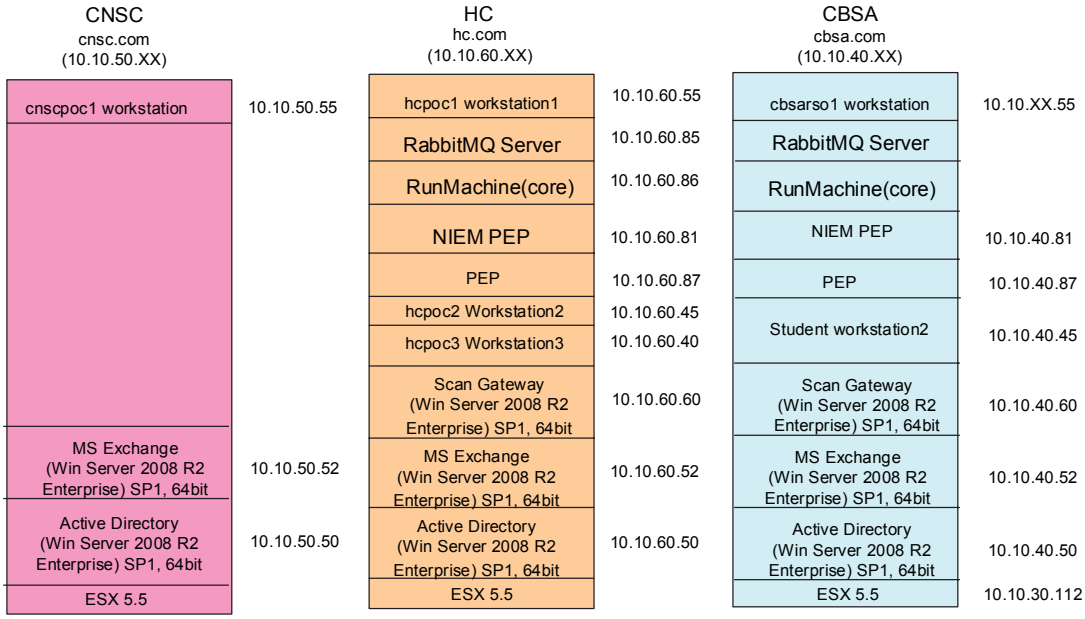

Figure 37 - Detailed Test Environment

The table below provides a listing of each user and their DCS security attributes:

Test Users

Table 17 - Test Users

Test Users		POLICY	
ID	PWD	COI	ACTION
CBSARSO1	Pass4Cord3	RAD,NUC,HEALTH CHEM	R/W=> permit R=>permit;W=>deny
Student	Pass4cord3	NONE	NONE
HCPOC1	Pass4Cord3	RAD,NUC,HEALTH	R/W=>permit
HCPOC2	Pass4Cord3	HEALTH	R=>permit
HCPOC3	Pass4Cord3	NUC	R=>permit
CNSCPOC1	Pass4Cord3	NONE	NONE

The diagram below provides an indication of the relationship between the various actors, their actions and the DCSS access controls for the file transfer processes.

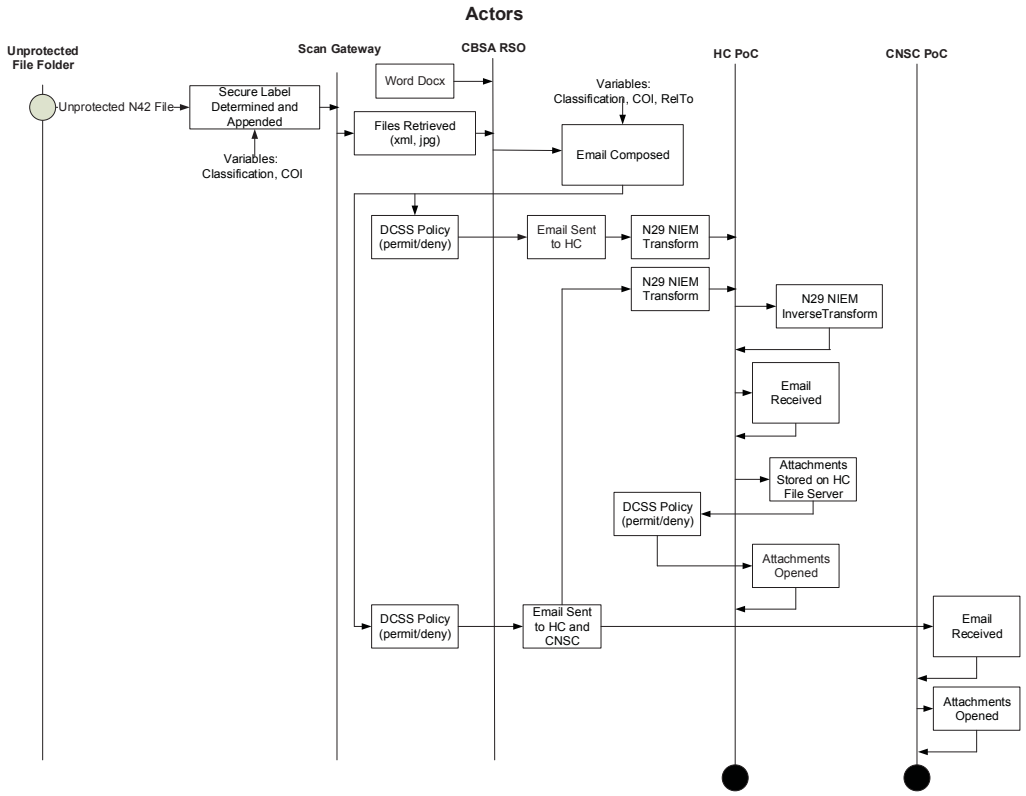

Figure 38 – Phase 3 Interactions

CBSA Secure File Labelling and Retrieval Test Scenarios

File Scenario Test 1 Auto Labelling

Overview: This scenario will demonstrate that an unlabelled file placed in an unprotected folder on the Scan Gateway Server will automatically be labelled and copied to the Scan Gateway DCS protected folder.

Actions: The CBSA RSO will place three files (1nuc1.n42, 2rad1.n42, and 1hc2.n42) into the unprotected folder (/home/aesir/testfiles). The DCS auto labelling tool (scannerLabel.py) will automatically create and append a security label to each of the files and copy the labelled files to the mounted and DCA "protected" folder on the Scan Gateway server.

File Scenario Test 2 DCS Permitted Retrieval of Files

Overview: This scenario will demonstrate the CBSA RSO1 viewing/listing the files in the protected folder and retrieving the files. The file security label data will be permitted by the DCS Policy for the user as a reader and writer.

Actions: The CBSA RSO1 user will list and retrieve the files from the DCS protected File Server and open the files, ensuring that the security label information and file metadata is correct.

File Scenario Test 3 DCS Denied View of Files

Overview: This scenario will demonstrate a Policy Violation by a non DCS controlled file services user attempting to view the files in the DCS protected folder.

Actions: A student user will attempt to view the files in the DCS protected folder.

Incident Reporting Tool Test Scenarios

This testing will verify that the IRT application data can be accessed by different users with different permitted/limited views based on their assigned DCS attributes.

IRT Scenario Test 1 Student Load files and View

Overview: This test will verify that the user student can load all files into the IRT database but only view a limited amount of data.
Actions: The student user within the cbsa domain, will through the browser interface, view a listing of files to be loaded, load the files, and observe the file data that the student user is permitted to view. The student user will log into the cbsa workstation2 machine and use the browser with the url http://10.10.40.87/irt to access the IRT database. The

loading of the files will be permitted and the student will have a limited view of the database records.

IRT Scenario Test 2 CBSA RSO View

Overview: This test will verify that the CBSA RSO user can view all rows, all records of the IRT database.

Actions: The CBSARSO1 user will log into the CBSA Win7 workstation and using the URL http://10.10.40.87/irt access the IRT database. The RSO will confirm that all files can be viewed.

CBSA OGD Email OGD Test Scenarios

Email Scenario 1 (Denied File)

Overview: A file user in a domain will send an email to another domain with an attachment. The attachment will contain a security label which is not releasable to another domain.

Actions: The CBSARSO1 in the CBSA domain will create an email message with CLASSIFICATION=SENSITIVE; and COI=CHEM and addressed to HCPOC1 in the Health Canada domain. The CBSARSO1 will attach a COI=CHEM security labelled document to the email. The email will not be received by HCPOC1 as the email and file is in the CHEM COI, which is not releasable from the CBSA domain.

The email will not be sent from the originating domain and the originator will receive a message alerting them to the fact that the email with the attachments could not be sent.

Email Scenario 2 (Permitted Files)

Overview: A user within a specific domain will create an email with attached files and have it sent to another government department's domain. The email and attached files, with associated metadata, will be viewable by permitted users in the other government department's domain.

Actions: The CBSARSO1 in the CBSA domain, using CLASSIFICATION=SENSITIVE; COI= RAD, NUC, and HEALTH will create an email message with permitted file attachments, and send it to the HCPOC1/2/3 users in the Health Canada domain. The HCPOC1/2/3 will receive the email message with the permitted attachments as links within the body of the email, plus the metadata associated with the attached files. The HCPOC1, HCPOC2, and the HCPOC3 users will be able to view the email message and open the permitted attachments through the links provided.

Email Scenario 3 (Non DCS Domain)

Overview: A user in the CBSA domain will create an email and attach the four protected files. The email will be sent to another government department, which does not have a DCS capability.

Actions: The CBSARSO1 in the CBSA domain will create an email with the four protected files as attachments and will send the email to the CNSC domain specifically to the user CNSCPOC1. The user CNSCPOC1 in the CNSC domain will receive the email with the attachments in the N.25 format including the metadata associated with each file.

Configuration File and Services locations:

The NIEM transformational files cbsaNiemPepSend, cbsaNiemPepRecv, and the cbsaNatPep are located in: /usr/local/bin

The email transactional and radmessage.n25 are located at: /etc/local/var/cbsa
Make sure the postfix mail service is started: sudo postfix start; and

The proxsmtp intercept is started using: sudo /usr/local/sbin proxsmtpd (-d 4 option can be used for debugging)

File Labelling Test Procedures and Results

File Labelling Test 1

Test No.1	Test Actions	Expected Results	P/F
	Actions	Results	
Auto Labelling	The CBSARSO1 will place three files (1nuc1.n42, 2rad1.n42, and 1hc1.jpg) into the unprotected folder (/home/aesir/testfiles).	On the CBSA PEP machine 10.10.40.87 run the ll command on the /home/aesir/testfiles and ensure that the files are listed.	Pass
	Within the /home/aseir/labeller folder run: sudo ./scannerLabel.py /home/aesir/testfiles /home/aesir/mount	The terminal display will indicate which file is being processed.	Pass
	Staying in the labeller folder run: cp ../testfiles/1nuc1.n42 ../testfiles/1nuc2.n42	The DCS auto labelling tool (scannerLabel.py) will automatically detect the new file created and append a security label to the file (as in 1nuc2.n42.zip) and copy the labelled files to the mounted and DCA “protected” folder on the Scan Gateway server.	Pass
	The CBSARSO1 client user on the Win7 machine will observer the newly created file in the DCS protected folder.	Within the U:data drive the new file will be listed	Pass

File Labelling Test 2

Test No.2	Test Actions	Expected Results	P/F
Permitted File Retrieval	Within Windows Explorer the CBSARSO1 client user on the Win7 machine will observe the DCS protected folder.	Within the W:/data drive the permitted files for the CBSARSO1 user will be listed.	Pass
	Within Windows Explorer the CBSARSO1 client user on the Win7 machine will right click on a protected zip file and left click on the “open” option. Select and open the customXml folder. Double click and open the item1.xml	Ensure that the Classification is SENSITIVE and the “Caveat” displayed is consistent with the file name.	Pass

File Labelling Test 3

Test No.3	Test Actions	Expected Results	P/F
DCS Denied View of Files	Using Windows Explorer the Student client user on the Win7 machine will attempt to observe the DCS protected folder.	Within the Z:/data drive the permitted files for the Student user will be listed. Ensure that no files are listed	Pass

IRT Test Procedures

IRT Test1

Test 1	Actions	Results	P/F
This test will verify that the user student can load all files into the IRT database but only view a limited amount of data	The student user within the cbsa domain, will through the browser interface, view a listing of files to be loaded, load the files, and observe the file data that the student user is permitted to view. The student user will log into the cbsa workstation2 win7 machine Open the Chrome browser and enter the url http://10.10.40.87/irt Login to	The IRT application will indicate that no files are loaded	Pass

	the IRT application as the student to access the IRT database.		
	The student user will left click on the Load File Button	Eight files will be shown: 167234 179687 180542 182354 184863 186796 188544 192398	Pass
	The student will left click on the load button associated with each file.	The loading of the files will be permitted and the student will have a limited view of the database records.	Pass
	The student will logout of the IRT application	IRT application returns to the Login screen	Pass

IRT Test2

Test 2	Actions	Results	P/F
This test will verify that the CBSA RSO can view all rows, all records of the IRT database.	The CBSARSO1 will, through the IRT application browser interface access the IRT database and view all file data. The CBSARSO1 user will log into the cbsa workstation win7 machine Open the Chrome browser and enter the url http://10.10.40.87/irt Login to the IRT application as the CBSARSO1 to access the IRT database.	The IRT application will indicate that all files are loaded and displayed: 167234 179687 180542 182354 184863 186796 188544 192398	Pass
	The CBSARSO1 user will left click on the Load File Button	Eight files with all details will be shown:	Pass
	The CBSARSO1 will left click on the load button associated with the file 167234.	The loading of the file will be permitted and the CBSARSO will have a complete view of all files loaded, including the latest 167234.	Pass
	The CBSARSO1 will logout of the	IRT application returns to the Login	Pass

	IRT application	screen	

OGD Email Test Procedures

Email Test 1

Test 1	Actions	Results	P/F
A user in the CBSA domain will send an email to a user in the HC domain with an incorrect attachment. The attachment will contain a chemical caveat security label, which the CBSA user does not have the policy rights to send out of the other domain.	The cbsarso1 within the cbsa.com domain, will send an email message with a chemical caveat and a file attached, which also has a chemical caveat to hcpoc1@hc.com	The sender cbsarso1 user will receive an "Undeliverable message" with a 550 permission denied statement. The hcpoc1@hc.com domain will confirm the email did not arrive.	Pass
Test1.1 No Security Label	The cbsarso1 user will send an email to hcpoc1@hc.com without a security label	The cbsarso1@cbsa.com will receive an email alert showing: 550 Email Label is missing. Cannot send message.	Pass

Email Test 2

Test 2	Actions	Results	P/F
A user within the CBSA domain will create an email with a number of attached files and have it sent to a user in the Health Canada domain. The email and attached files, with associated metadata, will be viewable by permitted users in the HC domain.	The cbsarso1 within the cbsa.com domain, will send an email message with a Radiation , Health, and Nuclear caveats and three attachments of Radiation (2rad3.n42.zip), Health (1hc5.n42.zip), and Nuclear (1nuc2.n42.zip) to: hcpoc1@hc.com, hcpoc2@hc.com, and hcpoc3@hc.com	Emails sent with no errors reported to cbsarso1.	Pass
	The HC user hcpoc1 will confirm receipt of the email and open all the attached files and confirm the metadata associate with each file event is correct.	Hcpoc1 can open the "filelinks.html" file and the metadata associated with each file is correct. 2rad3.n42.zip event metadata is: <metadata> <description>This is an N42 file.</description> <event_uuid>EVNT.8203.46</event_uuid> <subject_code>1457668203.46</subject_code> </metadata> 1hc5.n42.zip metadata is: <metadata> <description>This is an N42 file.</description>	Pass

		<event_uuid>EVNT.6978.59</event_uuid> <subject_code>1457976978.59</subject_code> </metadata> 1nuc2.n42.zip metadata is: <metadata> <description>This is an N42 file.</description> <event_uuid>EVNT.6875.83</event_uuid> <subject_code>1457976875.83</subject_code> </metadata>	
	The HC user hcpoc2 will confirm receipt of the email and attempt to open all the attached files.	The user hcpoc2 will only be able to open the link to the 1hc5.n42.zip file. The other two links will be Unauthorised and not opened.	Pass
	Confirm the event metadata for 1hc5.n42.zip is correct.	1hc5.n42.zip metadata is: <metadata> <description>This is an N42 file.</description> <event_uuid>EVNT.6978.59</event_uuid> <subject_code>1457976978.59</subject_code> </metadata>	Pass
	The HC user hcpoc3 will confirm receipt of the email and attempt to open all the attached files.	The user hcpoc3 will only be able to open the link to the 1nuc2.n42.zip file. The other two links will be Unauthorised and not opened.	Pass
	Confirm the event metadata for 1nuc2.n42.zip is correct.	1nuc2.n42.zip metadata is: <metadata> <description>This is an N42 file.</description> <event_uuid>EVNT.6875.83</event_uuid> <subject_code>1457976875.83</subject_code> </metadata>	Pass

Email Test3

Test 3	Actions	Results	P/F
A user in the CBSA domain will create an email and attach three DCS protected files. The email will be sent to CNSC, which does not have a DCS capability.	The cbsarso1 within the cbsa.com domain, will send an email message with a Radiation , Health, and Nuclear caveats and three attachments of Radiation (2rad3.n42.zip), Health (1hc5.n42.zip), and Nuclear (1nuc2.n42.zip) to: cnscpoc1@cnsc.com	The cnscpoc1@cnsc.com domain will confirm the arrival of the email with a “radmessage.N25” attachment, open the attachment and inspect the “radmessage.N25” to determine that the structure of the attachment is in accordance with the CNSC radmessage.n25.	Pass

Annex E – XSLT Primer

E.1 Introduction

The Canada Border Services Agency has a mandate to

> *"..work with law enforcement agencies to maintain border integrity and engaging in enforcement activities, including seizure of goods, arrests, detentions, investigations, hearings and removals."*

To further cooperation with other national and international partners, CBSA has conducted an examination the National Information Exchange Model (NIEM) as a potential standard to enable the flow of information exchange between CBSA affiliated networks. The intent is to enable protection on CBSA information assets by introducing data-centric security principles to gate access control to information assets and to ensure that the ingress and egress of CBSA information sets are in compliance with the organization's overarching security policy.

In this way, it is the current belief that by embracing the NIEM information standard,

- CBSA will achieve the required level of interoperability with other agencies; and
- Standardized expressions for information assets will simply the transition to and management of data-centric security protections.

One challenge to the adoption of NIEM is:

- There are many data standards that will need to be transformed into their NIEM equivalent; and
- The actual representation of the information within the NIEM constructs has not been definitely determined due to:
  - The evolution of the NIEM standard itself;
  - The maturing of several IEPDs that direct the representation of NIEM information for a specific information domain;
  - The need for further research into the correct method by which CBSA information should be encoded within NIEM constructs.

During the course of the investigation into the applicability of NIEM within the CBSA information architecture, a prototyping effort has demonstrated the validity of transforming CBSA radiological data in ANSI/IEEE N42.42 Standard to the N.25 Data Protocol.

- N42.42 is an industry supported, XML-based information formatting standard that allows manufacturers of radiation measurement instruments to interoperate with homeland security applications in order to support the detection of illicit trafficking of radioactive materials.

- N.25 was created to support the need for information-sharing in preventative radiological and nuclear detection for the US-based DHS Domestic Nuclear Detection Office (DNDO) and Customs and Border Patrol (CBP) agencies. During the development of N.25, a set of data elements not already existing in NIEM were developed as a common radiological and nuclear vocabulary and published by NIEM as the CBRN Domain.

However, in discussion with CBSA stakeholders, it was by no means certain that N.25 will be the representation of choice for CBSA NIEM data and, in any case, there will be additional source protocols other than N42 that will require a methodology to transition to their NIEM equivalent expression.

It was determined, therefore, that a general methodology to describe the approach that was taken to create the information transformation used in the prototype would be of benefit to future research and development activities. This section describes that general methodology.

E.2 Prototype Overview

The CBSA NIEM prototype was based on the simulated exchange of information sent from a CBSA domain to Health Canada (HC) and Canadian Nuclear Safety Association (CNSC) domains. The exchanges took the form of email exchanges where the email included an N.42 formatted attachment. A full description of the prototype architecture is provided in the *CBSA Data-Centric Security NIEM System Architecture Document & Prototype Report*. A brief outline of the exchange is defined below:

1. An originator in the CBSA domain sends the sample email from an Outlook email client to a recipient in one of the target domains (HC or CNSC).

2. The Exchange server routes the email through an outgoing NIEM Policy Enforcement Point (PEP). This NIEM PEP extracts the attachment from the email message.

3. The NIEM PEP applies an EXtensible Stylesheet Language Transform (XSLT) transformation on the email message's N.42 formatted attachment and creates a new file in NIEM/N.25 format.

4. This new file replaces the attachment on the outgoing message and the message is delivered to the destination domain.

5. The destination domain receives the NIEM/N.25 attachment as part of the email message.

It is the XSLT transformation that forms the core of this solution. It follows, then, that the methodology of how to use XSLT is key in that it will allow future efforts to be applied to any generalized data standard to data standard transformation.

E.3 XSLT described

With the adoption of XML as a standardized method for representing data in a structured fashion, it was quickly observed that there would be a need to translate one XML representation to another XML representation. A common requirement in this area is to allow generalized XML data structures to be presented in a XHTML for presentation to users through web-based interfaces. Just as common, however, is the need to transform the output from one information system into the input for another system; needs that frequently use XML as the language of interoperability.

This is the case for the NIEM prototype where the N.42 to N.25 NIEM information exchange is achieved through an XML to XML XSLT transformation.

XSLT transformations are performed by an XSLT processor, just as XML documents are read by an XML parser.

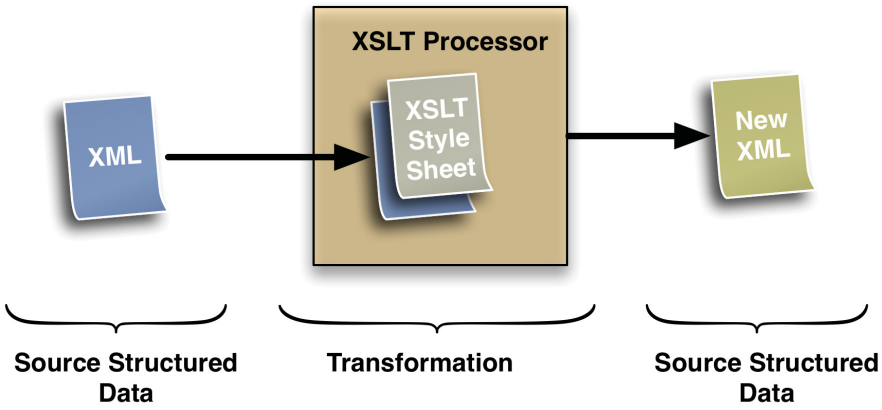

Figure 39 - An XSLT Transformation

The processor matches the original (input) XML file to a specified XSLT style sheet. XSLT style sheets are, essentiality, a set of coded rules that specify which data elements to extract from the input XML document, and how to transform the data to create a new output representation.

In expressing the transformation rules, an XSLT style sheet has two parts:

- a *pattern* which is matched against elements in the input XML; and
- A *template* that describes the format the result should take in the output file.

The XSLT processor starts the processing at the start of the XML data structure (the root element) and traverses the structure by:

- Matching data patterns to data nodes
- Extracting data from those nodes,
- Applying the template to create representative output structure.

This process repeats until the entire document has been processes and all rules have been applied as necessary. The resulting output data structure is then written out as the output file.

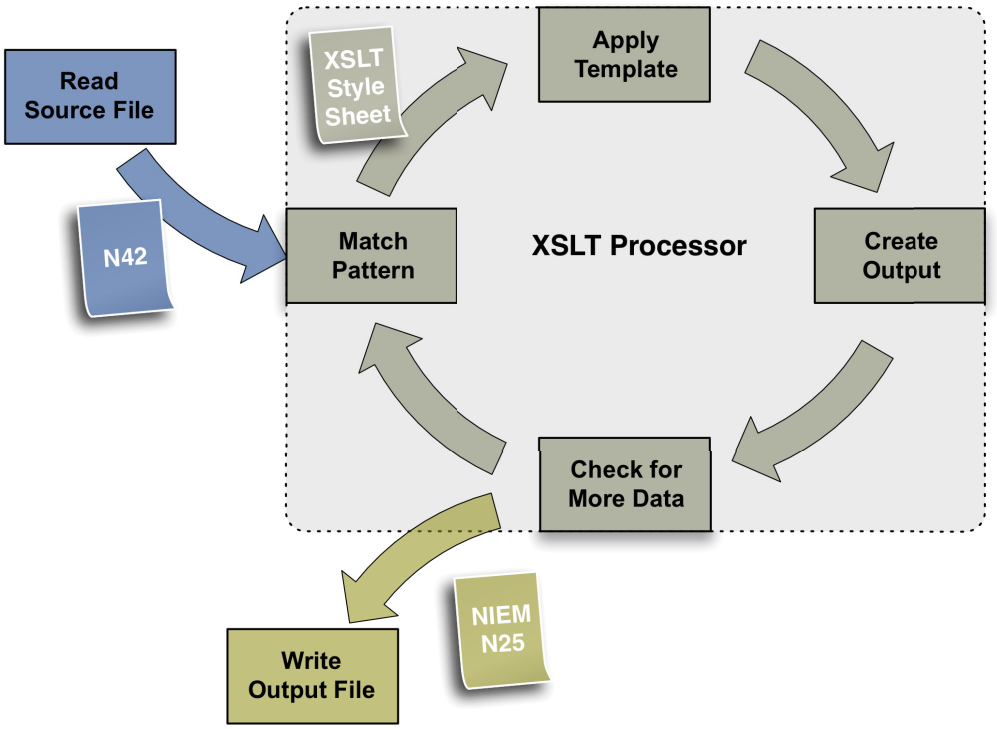

Figure 40 - XSLT Transformation Process

In XML parlance, the pattern matching function in XSLT transformations is done using XPath queries. XPath, like XSLT, is a part of the set of XML standards. It is a syntax for specifying and navigating to locations within an XML structured document. XPath expressions, therefore, can be used to uniquely identify locations within an XML document where specific XSLT transformation should take place.

Both XPath and XSLT have their own respective formatting specifications, syntax and processing logic. An understanding of these specifications is beyond the scope of this primer but is easily obtained from online resources, including:
http://www.w3.org/TR/xpath/ for information related to XPath and
http://www.w3.org/Style/XSL/ for information on the XSLT family of standards.

E.4 XSLT Methodology

In simplest terms, the process for generating an XSLT transformation follows this sequence:

1. Select the elements in the source XML that are needed to be part of the transformation.
2. Determine the desired representation of the data in the output format.
3. Identify the correct XPath expressions needed to extract the source elements from the original file.
4. Create the templates for that representation by coding the transformation using the XSLT style sheet syntax expressions.
5. Keep in mind how the XSLT processor will be used and how it will draw upon and create the source and target files, respectively

Both the creation of the query expressions and the use of XSLT style sheet language semantics are complex activities and should be done by individuals knowledgeable in generating this kind of code. Simple transformations can be developed quickly, however, more complex transformation not only require sophisticated use of the XSLT process, but are also vulnerable to incorrect coding which can lead to incorrect results.

E.4.1 XSLT Transformation Creation

XSLT is a rule-based and largely functional domain specific language. The absence of side-effects (situations where one transformation can affect another transformation within the same process) allows information managers to develop sets of XSLT rules in a completely incremental way.

For example, one can start with the identity transform, which simply maps the input to the output, and then add rules, testing the output after each addition secure in the knowledge that subsequent rules will not break work done earlier. This is a powerful development model and is very different from the development model used for procedural software.
Depending on the type of transformations that are required, the number and complexity of the rules could vary greatly.

If the transformation from the source file is to an identical tree (or to a tree that is a subset or superset of the input tree) in the target expression, then rules need only be

written to modify, trim or add elements. If the transformation is from one tree structure to another, however, then the processing rules can be very complex requiring an XSLT developer to keep in mind both the input and output trees simultaneously.

Regardless of the difficulty of the transformation, individual XSLT rules may have to look ahead in the input tree structure (to create a table of contents for example), or to look back in the input data (to create summary data like an index for example). This is possible because the XML parser that fronts the XSLT processor will have stored the input document internally as a tree structure through which the XSLT rules can wander freely.

XSLT scripts can map:

- A single input document to a single output document (for example, add a list of figures to the document),
- Many input documents to one output document (for example, to create a summary document like a magazine from a collection of articles), or
- One input document to many output documents (for example to transform an XML book document into a collection of hyperlinked XHTML output documents).

E.4.2 XSLT Mapping Challenges

There are many challenges in mapping between XML documents; which is why it requires a trained professional in information management to create, validate and maintain XSLT transformation code.

Mapping between XML documents can be easy or difficult depending on the nature of the problem. Not all mappings will even be possible, but as will be seen below, this is a failure of the nature of the specification of the information management problem, not due to limitations of XSLT itself.

E.4.2.1 Managing Document Structures

It can be difficult to map between documents with different tree structures (or shapes). Content in the input may have to be broken up and reordered for output. The output document may simply not have a place to hold parts of the input content. In this case, a developer may have to drop input data or store it in a useful (but possibly inappropriate place) in the output. For example, input data may have to be stored in the output as a comment along with special notes to let the output consumers know about the

significance of certain comments. Data loss may or may not be a problem depending on what the document transformation is intended to do.

A more subtle problem occurs when an output document requires an element, element content, or subtree and the input document does not contain relevant equivalents. XSLT rules can inject the required output easily enough, but anything injected like this is dangerous as it is indistinguishable from real content.

Mapping fidelity and processing pipelines must also be considered when creating document mappings. For example, if a document will be mapped to an intermediate format (for transport or storage for example) and is always then mapped back into the original format, it may be acceptable to allow the intermediate format to be invalid (according to its document model). For this approach to work, however, both the mapping and unmapping must be controlled by individuals that understand the compromises being made.

E.4.2.2 Managing Data Types

With the advent of XML Schema, and similar document modeling languages, XML documents can draw upon a wide range of data types. This can be a significant problem if the input document is does not use a large range of data types or, worse, uses data types in a different manner than expected.

For example:

- *String* input may have to map to an enumerated type. If the input string does not contain one of the values in the output enumerated list, the output document will not validate.
- Input values may specify values in a range that is not valid in the output document.
- Input date formatting may be ambiguous (Y/M/D or Y/D/M) or incomplete (no year for example) as far as the output format is concerned.

E.4.2.3 Attributes vs. Elements

Similar to data typing, XML element content and attribute content is roughly logically equivalent and so it should be possible to convert between simple textual element content and attribute values. However, different syntax rules for what is allowable as element content or attribute content may make conversion difficult in practice.

E.4.2.4 Semantic

In most cases, the syntactical rules needed to map an input to an output can be determined. In some cases, however, a subject matter expert (SME) may have to be involved to understand the actual meaning behind the source and target expressions to ensure that this meaning is maintained through the transformation. It is frequently seen that the transformations that require the most 'domain specific' knowledge are the transformations that are the most important to the organization. If the XSLT author does not have this domain knowledge, consultation with a domain SME is usually required.

E.4.2.5 Other challenges

The processes that create a document type may not be adequate to allow the documents to be transformed into a different type. For example, documents may not have enough metadata attached or available from their creation environment to allow a complete mapping into a different document type. When this situation is encountered it usually reflects the fact that there is a more significant underlying information management concern.

XSLT is a rule-based and mostly functional programming language. There are not as many programmers familiar with the language or programming model that can make acquiring competent information management resources difficult.

E.4.3 XSLT Tools

Since creating XSLT programs can be a cumbersome and error prone activity for all but the simplest of transformations, there exist free and commercial tools that can assist and automate the generation of these programs. Two notable examples of this class of tool are:

- Altova MapForce (http://www.altova.com/mapforce.html) and
- Liquid XML Data Mapper (http://www.liquid-technologies.com/xmldataMapper.aspx)

Mapforce is particularly useful for automating the process as it not only aids in the mapping of source to target expression but will also generate the actual resulting XSLT style sheet.

The open source tools available for processing XSLT are limited to XSLT v1.0. XSLT is now at V3.0 so if there are any features of XSLT versions 2 or 3 essential to a transformation,

proprietary tools will be required. It is possible to work around this limitation with a hybrid processing model where the bulk of a transformation is expressed in XSLT and the remainder of the work is done using programmatic access to the document tree created by the XML parser.

Annex F – Secure NIEM Exchange Investigation

F.1 Purpose

This document provides the results of the investigation work carried out on the various XML international standards involved in information exchange and data security.

F.2 Background

F.2.1 US Government NIEM Activity

NIEM in the U.S. is a partnership of the U.S. Department of Justice and the Department of Homeland Security. In a memorandum signed on 28 March 2013, the DoD Chief Information Officer (CIO) announced that DoD will adopt NIEM as the basis for its data exchange strategy in coordination with the NIEM Program Management Office (PMO). NIEM developers recently announced an interim solution designed to allow users to use Intelligence Community (IC) Information Security Markings (ISM) within NIEM 2.0. It is one of the IC Metadata Standards for Information Assurance and is the preferred way to apply information security markings within XML instances. Until recently, the schema for the Intelligence Community Information Security Marking (IC-ISM) standard was considered for official use only (FOUO) and could not be published. Therefore, NIEM 2.0 could not integrate components of IC-ISM without publishing the IC-ISM schema. Actions have now been taken to restore the ability to use IC-ISM within NIEM 2.0 and future releases.

F.2.2 NATO

Within NATO there are two specific standards, which are under development, and are relevant to the securing of information exchanges. They are STANAG 4774 Confidentiality Metadata Label Syntax and STANAG 4778 Metadata Binding Mechanism

F.2.3 NIEM

NIEM domains contain mission-specific data components that build upon NIEM core concepts and add additional content specific to the community supporting that mission. NIEM's Intelligence domain is the standard of choice for exchanging intelligence among any federal, state, and local agencies on a foreign or domestic basis. Domain members identify the operational needs to exchange intelligence, as well as the opportunities to share information with other domains and functions in justice and homeland security. Domain members represent the full range of operations that deal with the gathering, analysis, fusion, and dissemination of intelligence, as well as the ability to act upon it.

The NIEM intel domain leverages the three data specifications from the Director of National Intelligence, Intelligence Community, which deal with metadata security and information sharing, namely:

- IC-TDF - The Data Encoding Specification for Trusted Data Format(IC-TDF.XML) defines detailed implementation guidance for using Extensible Markup Language (XML) to encode IC-TDF data;
- IC-ISM - The Data Encoding Specifications for Information Security Marking Metadata defines XML elements and attributes, associated structures and relationships, mandatory and cardinality requirements, permissible values, and constraint rules for representing electronic information security markings; and
- IC-NTK - The Data Encoding Specification for Need-To-Know Metadata defines the XML elements and attributes; associated structures and relationships; mandatory and cardinality requirements; and permissible values for representing NTK metadata associated with an information resource or part of an information resource using XML.

F.3 Standards Reviewed

The U.S. Government, NATO, and NIEM are active in information exchange security standards with respect to binding and integrity of security metadata to data.

F.3.1. U.S. Government

Within the U.S. Government the Director of National Intelligence, has a number of data specifications with respect to the binding, and integrity of security label metadata to data. Table 18 is a listing of all the data encoding specifications for information being shared or exchanged within an enterprise. Highlighted in bold text are the data encoding specifications, which are considered relevant to this investigation.

Table 18 - U.S. Government Data Encoding Specifications

Data Encoding Specifications
Abstract Data Definition
Access Rights and Handling
Authority Categories
Community Shared Resources
Document and Media Exploitation
DoD Discovery Metadata
Enterprise Audit
Enterprise Data Header
Fine Access Control
Geopolitical Entities, Names, and Codes

Information Resource Metadata
Information Security Marking Access
Information Security Marking Country
Information Security Marking Metadata
Intelligence Publications
Intelligence Community Access Control
Intelligence Community Identifier
Intelligence Community Only Need to Know
ITS - Organization Messaging
Multi Audience Collections
Multi Audience Tearline
Need-To-Know Metadata
ORCON Need-To-Know Access
Revision Recall
Trusted Data Format
US Agency Acronym
US Government Agency
Virtual Coverage

F.3.1.1 Need-To-Know Metadata

The XML Data Encoding Specification for Need-To-Know Metadata defines the XML elements and attributes; associated structures and relationships; mandatory and cardinality requirements; and permissible values for representing NTK metadata associated with an information resource or part of an information resource using XML. NTK.XML can be incorporated into other Data Encoding Specifications. NTK metadata facilitates automated systems making a “need-to-know” (NTK) access determination about an information resource. These metadata are used to represent the system-specific properties assigned to an information resource that will be used, in conjunction with information about the user, and possibly other information, to determine the user’s access to the data. A single information resource may include multiple occurrences of these metadata in order to specify NTK information according to multiple, different access systems. Compliance with this specification is measured against all aspects of the technical and documentary artifacts contained within the specification release package. The specification is maintained by the IC Chief Information Officer via the Data Coordination Activity (DCA) and Common Metadata Standards Tiger Team (CMSTT).

F.3.1.2 The Trusted Data Format

This XML Data Encoding Specification for Trusted Data Format (IC-TDF.XML) defines detailed implementation guidance for using XML to encode IC-TDF data. The IC Trusted Data Format XML specification is the IC submission format for binding assertion metadata

with data resource(s). This TDF functionality supports the IC way ahead strategy of implementing secure cloud-based information exchange and discovery on the IC Enterprise. The TDF elements, complex types, simple types, attributes, and element groups are shown in Table 19.

Table 19 - Trusted Data Format

Elements	Complex Types
AssertionGroup/Assertion	AssertionType
AssertionGroup/HandlingAssertion	AttachedKeyType
AssertionType/StatementMetadata	Base64BinaryValueType
AttachedKeyType/KeyValue	BindingType
BindingGroup/Binding	BoundValueListType
BindingGroup/ReferenceList	BoundValueType
BindingType/BoundValueList	EncryptionMethodType
BindingType/SignatureValue	HandlingAssertionType
BindingType/Signer	KeyAccessType
BoundValueListType/BoundValue	PasswordKeyType
EncryptionInformationGroup/EncryptionInformation	PreSharedKeyType
EncryptionInformationGroup/EncryptionInformation/EncryptionMethod	ReferenceListType
EncryptionInformationGroup/EncryptionInformation/KeyAccess	ReferenceType
EncryptionMethodType/KeySize	ReferenceValueType
EncryptionMethodType/OaepParams	RemoteKeyType
HandlingAssertionType/HandlingStatement	SignatureValueType
KeyAccessType/AttachedKey	StringValueType
KeyAccessType/PasswordKey	StructuredValueType
KeyAccessType/PreSharedKey	TdcType
KeyAccessType/RemoteStoredKey	TdoType
KeyAccessType/WrappedKey	WrappedKeyType
PayloadGroup/Base64BinaryPayload	
PayloadGroup/ReferenceValuePayload	
PayloadGroup/StringPayload	
PayloadGroup/StructuredPayload	
ReferenceListType/Reference	
StatementGroup/Base64BinaryStatement	
StatementGroup/ReferenceStatement	
StatementGroup/StringStatement	
StatementGroup/StructuredStatement	
TrustedDataCollection	
TrustedDataObject	
WrappedKeyType/EncryptionMethod	

WrappedKeyType/KeyValue	

Simple Types	Attributes
CVEnumTDFAppliesToState CVEnumTDFHashAlgorithm CVEnumTDFSignatureAlgorithm MediaTypeType	@filename @id @idRef @includesStatementMetadata @isEncrypted @mediaType @normalizationMethod @scope @version AssertionType/@type AssertionType/StatementMetadata/@appliesToState BindingType/Signer/@issuer BindingType/Signer/@serial BindingType/Signer/@subject BoundValueType/@hashAlgorithm EncryptionInformationGroup/EncryptionInformation/@sequenceNum EncryptionMethodType/@algorithm HandlingAssertionType/@appliesToState PasswordKeyType/@algorithm PreSharedKeyType/@alias PreSharedKeyType/@store ReferenceValueType/@uri RemoteKeyType/@protocol RemoteKeyType/@uri SignatureValueType/@signatureAlgorithm WrappedKeyType/@keyIdentifier

Element Groups
AssertionGroup BindingGroup EncryptionInformationGroup PayloadGroup StatementGroup

F.3.1.3 Information Security Marking Metadata

This IC enterprise data encoding specification defines XML elements and attributes, associated structures and relationships, mandatory and cardinality requirements, permissible values, and constraint rules for representing electronic information security markings. The standard supports Executive Order (EO) 13526, Classified National Security Information which "prescribes a uniform system for classifying, safeguarding, and declassifying national security information", across national security disciplines, networks, services, and data.

This standard is a technical bridge between:

- Security marking requirements defined by the National Archives and Records Administration (NARA)/Information Security Oversight Office (ISOO);
- IC security markings register maintained by the Office of the Director of National Intelligence (ODNI)/Controlled Access Program Coordination Office (CAPCO); and
- Information technology solutions that implement structured security marking metadata.

This specification changed names and numeric designators multiple times since its inception in the late 1990's. Each version listed below supersedes the previous version. The IC Chief Information Officer maintains this specification via the Data Coordination Activity (DCA) and Common Metadata Standards Tiger Team (CMSTT).

Information sharing within the national intelligence enterprise will increasingly rely on information assurance metadata (including information security markings) to allow interagency access control, automated exchanges, and appropriate protection of shared intelligence when necessary.

A structured, verifiable representation of security marking metadata bound to the intelligence data is required in order for the enterprise to become inherently "smarter" about the information flowing in and around it. Such a representation, when implemented with other data formats, improved user interfaces, and data processing utilities, can provide part of a larger, robust information assurance infrastructure capable of automating some of the management and exchange decisions today being performed by human beings.

Throughout the intelligence life cycle, the enterprise needs:

- User interfaces and processing logic that helps users and services to reliably assign and manipulate information security markings at the portion and document level;
- Automated rendering of electronic portion markings, security banners, classification authority blocks, and other security control markings in accordance

with the IC's classification and control marking system and associated executive orders, statutes, and DNI policies;

- Marking validation to ensure controlled values and business rules are followed; and
- Cross-domain discovery, access, and dissemination capabilities based on access policy logic that leverages electronic security markings along with other key metadata about users, services, clearances, and access environments.

F.3.2 NATO

Within NATO there are two specific standards, which are under development, and are relevant to the securing of information exchanges. They are STANAG 4774 Confidentiality Metadata Label Syntax and STANAG 4778 Metadata Binding Mechanism.

F.3.2.1 NATO STANAG ADatP-4774 Confidentiality Metadata Label Syntax

The NATO STANAG ADatP-4774 Confidentiality Metadata Label Syntax, provides common XML-based formats and syntax for security policies and confidentiality metadata. Information objects and data assets can be labelled to support access and release decisions in a manner that is understandable to all partners.

The five principles adopted by the ADatP-4774 are:

- Information Ownership and Custodianship. Information shall have an originator, and clearly defined ownership and custodianship assigned throughout its life-cycle;
- Information Sharing. Information shall be managed with an emphasis on the 'responsibility-to-share' balanced by the security principle of 'need-to-know', and managed to facilitate access, optimise information sharing and re-use, and reduce duplication, all in accordance with security, legal and privacy obligations;
- Information Standardisation. Information shall have standardised structures and consistent representations to enable interoperability, cooperation and more effective and efficient processes;
- Information Assurance. Information shall be protected by applying the principle of Information Assurance, which is described as the set of measures to achieve a given level of confidence in the protection of communication, information and other electronic systems, non-electronic systems, and the information that is stored, processed or transmitted in these systems with respect to confidentiality, integrity, availability, non-repudiation and authentication; and
- Data Assurance. The authority of the source and integrity of the data can be determined and assessed because of the history, security level, and access control level of each data asset is known and available.

The details of the ADatP-4774 xml structure, in terms of elements, types, and attributes are provided in Table 20.

Table 20 - NATO 4774 Structure

Elements	Complex Types
Category	ConfidentialityLabelType
CategoryValue	ConfidentialityLabelBaseType
GenericValue	ConfidentialityInformationType
IntegerValue	PolicyIdentifierType
BitStringValue	ClassificationType
OriginatorID	CategoryType
CreationDateTime	OriginatorIDType
SuccessionHandling	SuccessionHandlingType
SuccessionDateTime	
SuccessorConfidentialityLabel	

Simple Types	Attributes
RequiredToken	Id
PrivacyMarkType	ReviewDateTime
CategoryValueType	URI
GenericValueType	Type
IntegerValueType	TagName
BitStringValueType	IDType
CreationDateTime	
SuccessionDateTimeType	

F.3.2.2 NATO STANAG ADatP4778 Metadata Binding Mechanism

The objective of this standard is to describe generic concepts for binding of metadata to data objects for use within a specific domain or enterprise, and that can easily be interpreted and processed among a federation of enterprises. The standard does this by providing a formal and consistent way to describe and categorise binding mechanisms of various types and strengths. The standard defines three approaches for binding metadata with data objects:

- Encapsulating: The data object together with the metadata is encapsulated within the Binding and is represented by a new composite data object. For example, with eXtensible Markup Language (XML) the use of Binding as a new data object as its root element with the data object and metadata contained directly within the binding element;

- Embedded: The binding information is embedded within the data object and the Binding contains a reference to the information. For example, an XML data object may use a schema that either includes a binding element, or allows it to be extended with arbitrary elements; and
- Detached: The metadata may be stored in a separate structure from the data object with the two linked by reference. However, the binding information and the data objects are always detached. For example, a separate file containing the metadata and the binding element references an XML or JPEG file within a file system via a URI.

The standard states, which binding approach to choose depends on many factors and therefore this standard cannot prescribe if and when to best apply a particular binding approach. The standard further states "A cryptographic binding (that includes cryptographic artefacts) uses cryptographic techniques and mechanisms like cryptographic digests, message authentication codes or digital signatures in order to protect the binding. Such cryptographic techniques and mechanisms are subject to the level of assurance required for protecting the integrity of the binding and for establishing confidence for the authenticity of the entity creating the binding. The level of assurance required for protecting the integrity of the binding and for establishing confidence for the authenticity of the entity creating the binding is a matter for organizational, national or federation security policies."

The standard through METADATA BINDING PROFILES indicates how the Binding Information is applied to specific data formats and protocols, including the following:

- Web Services (SOAP-based and REST-based web services);
- Informal messaging (SMTP/MIME internet email);
- Collaboration (Text-based instant messaging);
- Document management (including Office Tools); and
- Arbitrary Files.

The standard references two RFC's with respect to Email:

- IETF RFC 7444, "Security Labels in Internet Email", K. Zeilenga and A. Melnikov, February 2015; and
- IETF RFC 5322, "Internet Message Format", at http://tools.ietf.org/html/rfc5322, October 2008.

There is linkage to the NATO 4774 Confidentiality Metadata Label standard. Of specific interest within the CBSA experiment is Annex C of NATO 4778, the Metadata Binding associated with SMTP/MIME emails.

F.3.3 NIEM Intelligence Namespace (Intel)

In NIEM 3.0 the augmentations methodology has been updated. Augmentations are a way that a namespace, in a particular domain, can define an extension to a base type without needing subclasses. For example the NIEM Core person type for which Immigration and Intel need to have additional information they convey about the person. Now what they don't need to do is to find a new special type of person that is an extension of a base person type. Instead, these are additional characteristics of a person that are specific to a domain. And the method for doing this now is a type, such as the person type defines a person augmentation point. The name of the element is constrained to be the same as the name of the type but with the word type being replaced by augmentation point, which is an element that has no type and is set to be abstract. Being "typeless" means that elements of various types can be substituted for it. Being abstract means that the element itself is not allowed to appear in an instance so it can't act as a wildcard. Any domain that needs to find an augmentation defines that element of an augmentation type and sets it to be substitutable for the augmentation point.

The structures base types have been refactored. Complex object type has been renamed; it contains an augmentation point which is defined as structures augmentation point, which has the same qualified name as the type but with the word type replaced with the word augmentation point. So you could substitute in for object augmentation point if you wanted to define something that has said additional stuff about any given object. The structures: ref, has been added, which provides the reference element or the flexible content of content or reference. Link meta data has been renamed to relationship meta data, and last the addition of a wildcard for IC ISM and NTK namespaces. These are defined by the Intelligence Community as a Trusted Data Format which includes the ISM and NTK. ISM is for security markings and NTK is for need-to-know. If there is a need to work inside a TDF environment, this should provide all that is needed to support the required markup inside payloads of TDF messages.

It would appear that the IC-ISM XML Schemas will not be included as part of the NIEM 3.0 release. Instead, each of the NIEM structures base types (ObjectType, AssociationType, AugmentationType, and MetadataType, as well as SimpleObjectAttributeGroup) incorporates an anyAttributes element as part of its definition. This wildcard allows any attributes to appear that have the IC-ISM namespace, as well as those having the IC-NTK namespace. This is in line with the recommendations of the developer of the IC Trusted Data Framework, and should support ISM, NTK, and the TDF.

Table 21 provides a list of the elements and complex types defined in the NIEM Intel 3.0 xsd file.

Table 21 - Elements & Complex Types in NIEM 3.0

Elements	Complex Types
AgencyInterestCategory	AgencyInterestCategoryType
AgencyInterestCategoryAugmentationPoint	AgencyInterestOtherCategoryType
AgencyInterestCategoryCodeText	AgencySubjectHandlingType
AgencyInterestCategoryDescriptionText	AgencySubjectInterestType
AgencyInterestCategoryText	BiometricAugmentationType
AgencyInterestOtherCategory	CapabilityAugmentationType
AgencyInterestOtherCategoryAugmentationPoint	ContactInformationAugmentationType
AgencyName	DayType
AgencySubjectHandling	IdentificationAugmentationType
AgencySubjectHandlingAugmentationPoint	IdentityAssociationAugmentationType
AgencySubjectHandlingFBICodeText	LocationAugmentationType
AgencySubjectInterest	PersonAugmentationType
AgencySubjectInterestAugmentationPoint	PersonCitizenshipDetailsType
AuthenticIndicator	PersonEducationDetails
AuthenticatedIdentity	PersonInIDType
BiometricAugmentation	PersonOtherIDType
CBEFFText	PotentialIdentityMatchAssociationType
CapabilityAugmentation	SubjectCautionInformationType
CapabilityProficiencyText	SubjectHandlingType
ContactInformationAugmentation	
ContactSatelliteTelephoneNumber	
ConveyanceRegistrationIdentification	
DayDate	
FBIECRCode	
GlobalRegionCodeText	
IdentificationAugmentation	
IdentificationIssuingCountry	
IdentificationIssuingCountryFIPS10-4Code	
IdentificationIssuingCountryISO3166Alpha2Code	
IdentificationIssuingCountryName	
IdentificationIssuingLocalityText	
IdentificationIssuingStateName	
IdentityAssociationAugmentation	
LocationAugmentation	
LocationCountryFIPS10-4PlusNCTCCodeText	
MonthDate	
OccupationName	
PersonAffiliationAssociation	
PersonAugmentation	
PersonCauseOfDeathText	

PersonCitizenshipDetails PersonCitizenshipDetailsAugmentationPoint PersonCitizenshipStatusCodeText PersonContactDetails PersonEducationDegreeCodeText PersonEducationDetails PersonEducationDetailsAugmentationPoint PersonEyeDescriptionText PersonInIDAugmentationPoint PersonInIdentification PersonLocationUsageCodeText PersonMultimediaIDBinary PersonOtherIDAugmentationPoint PersonSystemIdentification PhysicalFeatureCategoryCodeText PotentialIdentityMatchAssociation PotentialIdentityMatchAssociationAugmentationPoint PotentialIdentityMatchDescriptionText PrimaryIdentity StateINACodeText SubjectCategoryCodeText SubjectCautionInformation SubjectCautionInformationAugmentationPoint SubjectCautionInformationDescriptionText SubjectHandling SubjectHandlingAgencyName SubjectHandlingAugmentationPoint SubjectHandlingCodeText SubjectHandlingDescriptionText SubjectReasonOnListText SystemIdentification ThreatCategoryCode	

To summarize - there is no security label metadata indicated in the NIEM 3.0 schema. There appears to be an integration of the NIEM 3.0 Intel namespace and the security metadata associated with the IC ISM, NTK, and TDF namespaces through the augmentation, specified base elements and the specific" anyAttribute" element.

F.4 Recommendation & implementation strategy

The U.S. IC-ISM, IC-TDF and IC-NTK specifications are highly developed and provide comprehensive coverage of the elements, types, and attributes required to provide a level of assurance for electronic information security markings and binding of security metadata to data. The richness of the specifications may be overly complex for the current needs of the investigation into information exchange using NIEM. The adoption and number of departments and agencies within the U.S. actually using these specifications for information exchange is currently unknown.

The NIEM 3.0 Intel namespace makes use of the IC-ISM, IC-TDF and IC-NTK data specifications, by the use of the core anyAttributes element, to provide the electronic information security markings and binding of security metadata to data.
The Bell/Cord3 research and development team is familiar with the NATO STANAG work having participated in a number of NATO STANAG 4774 and 4778 workshops over the last two years. Therefore, the approach taken in the CBSA prototype is to adopt the simpler, and more concise, NATO STANAG 4774 Confidentiality Metadata Label Syntax. This security metadata will be included within the NIEM IEPD N.25 Radiation Device Messaging. This approach allows a single security labeling standard to be used throughout the CBSA prototype, including both within the CBSA domain and for information exchanged with OGDs.

Within DCSS the binding of label security metadata to the data is achieved through the use of symmetrical encryption. In addition, secure hash algorithms and techniques are employed to determine if a security label associated with a file or document has been tampered with. This level of “loose” binding (i.e. not a digital signature) is deemed to provide a sufficient level of assurance within a security domain. However, this approach does not necessarily work well when exchanging information between security domains. Consequently, S/MIME will be used to encrypt the data, and strongly bind the security label to the data, as it transits between security domains.