SAMSON Technology Demonstrator

Final Report

Prepared by:
Bell Development Team
Bell Canada, 160 Elgin St., 17th Floor
Ottawa, ON K1S 5N4

PWGSC Contract Number: W7714-08FE01

CSA: D. Charlebois, DRDC, Ottawa Research Centre

Defence Research and Development Canada

Contract Report
DRDC-RDDC-2014-C101
June 2014

DEFENCE R&D DÉFENSE

Defence Research and Development Canada

SAMSON Technology Demonstrator
Final Report
Phase IV
PM007

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

Final
Oct 2013

Table of Contents

1.0 INTRODUCTION 3
1.1 DATA-CENTRIC SECURITY 4
1.2 PROJECT APPROACH 6
2.0 SAMSON DESIGN STRATEGY 9
2.1 SAMSON MESSAGING INFRASTRUCTURE 10
2.2 SECURITY SERVICE GATEWAYS 12
2.3 POLICY ENFORCEMENT POINTS 15
3.0 RESULTS 20
3.1 SAMSON DEPLOYMENTS 20
3.2 TESTING RESULTS 21
3.3 CERTIFICATION AND ACCREDITATION PROCESS 22
4.0 FUTURE WORK 23
4.1 ACTION PLAN 26
4.2 RESEARCH EFFORT PRIORITIES 29
5.0 CONCLUSIONS 31

Table of Figures

Figure 1: SAMSON Capability Model 3
Figure 2: The SAMSON Information Protection Model 5
Figure 3: The SAMSON Security Overlay 7
Figure 4: SAMSON Core Components 9
Figure 5: A Security Service Gateway (SSG) 12
Figure 6: PEP Data Mediation Process 16
Figure 7: Future SAMSON Work 25
Figure 8: Prioritized Future Work Items 30

Table of Tables

Table 1: Action Responses 26
Table 2: Future Work Action Plan 28
Table 3: Research Priorities 29

1.0 Introduction

The Secure Access Management for a Secure Operational Network (SAMSON) Technology Demonstrator (TD) project demonstrates the integration of a data-centric security protection model into existing operational environments.

This integration of data-centric security is achieved though the delivery of the following capabilities that apply object-level security protection to individual information assets:

1. Access Management: Security labels attached to individual information assets determine what actions can be performed against each asset.

2. Information Protection: Each information asset is uniquely encrypted.

3. Auditing: Actions performed against each information asset are recorded in a trusted, tamper-resistant audit trail.

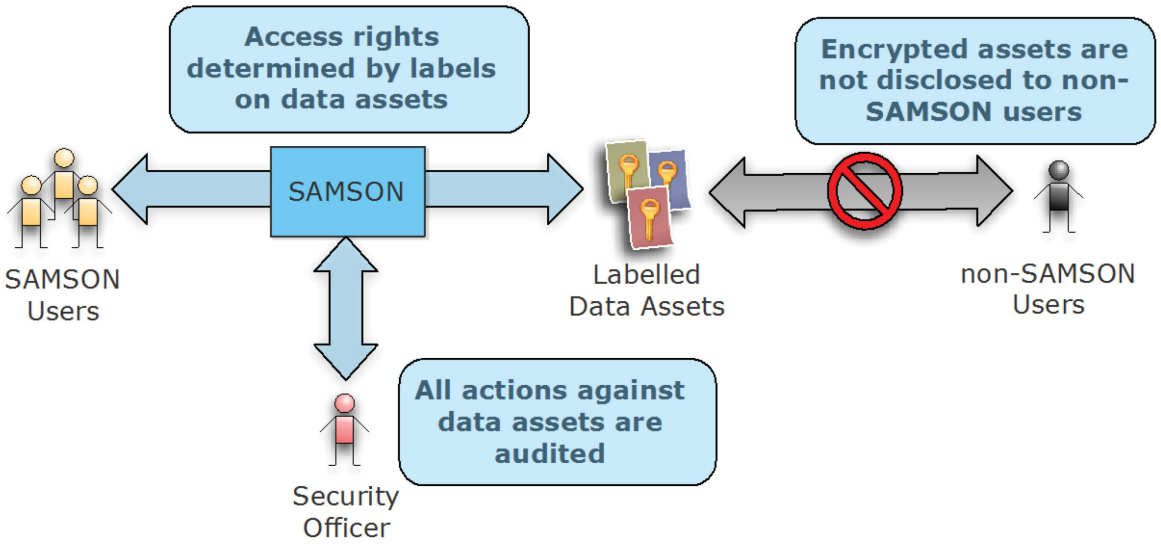

Figure 1: SAMSON Capability Model

These three core capabilities form a reliable foundation that enables the controlled sharing of information assets between trusted parties while ensuring that information is not disclosed in violation of access control policies. Information assets, in the SAMSON context, can be any data type or service. While the SAMSON TD demonstrated object-level protection of files, email messages, instant messages and web sessions, there is no limitation on the types of data that can be protected through the SAMSON data-centric security model.

1.1 Data-Centric Security

In the context of this project, data-centric security is an architectural approach to securing information assets by using the security attributes attached to individual data assets to determine and implement the appropriate level of security for that asset. Universally in data-centric security modelling, the attributes associated with data are a reflection of that asset's value. That is, the value of the asset to the organization and the impact that improper protection of that asset will have on the organization's security posture.

The SAMSON Information Assurance (IA) protection model uses security metadata bound to data assets in order to apply the appropriate security mechanisms for the protection of that asset. Within the Government of Canada (GoC) and Department of National Defence (DND) milieu, the attributes associated with an information asset are represented within the asset's *security label* that comprises:

- The security metadata; and
- The binding mechanism that ties the label to the data.

The security label, frequently expressed as a mark-up for human readable documents, includes: policy identifiers, classification and release categories (caveats)[1].

Access management, in the SAMSON information protection model, leverages three constructs to exert control over the release of data:

1. The caveats on individual data assets;
2. The communities of Interest (COI) to which users belong; and
3. The unified and holistic security policy that determines what rights users and groups have over specific caveats.

These three constructs mean that information owners and security officers have three degrees of freedom when determining who can access sensitive information:

[1] Much of the SAMSON security labelling standard is taken from documents published by the NATO C3 Agency, specifically "A Proposal for an XML Confidentiality Label and Related Binding of Metadata to Data Objects"

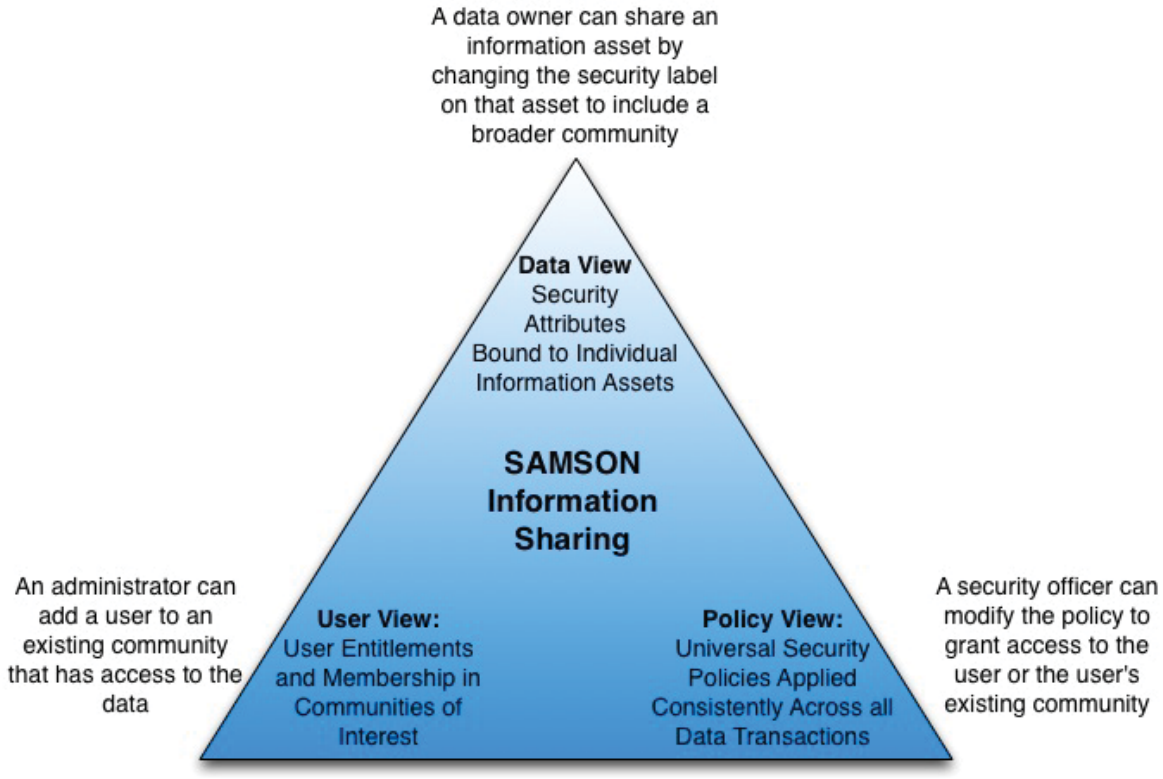

Figure 2: The SAMSON Information Protection Model

It is the assertion of the SAMSON project that when IA can be assured using a data-centric security model, it is possible to collapse multiple SECRET networks into a single operational network where separation of caveats is done on individual information assets.

In execution, the objective of the SAMSON TD is to introduce caveat separation within a single operational network such that all data objects are subject to the same security protections, including:

- How access to those data assets is granted;
- How transactions against those data assets are controlled through policy; and
- How those data assets are protected against disclosure.

These same protections are applied across all data assets regardless of the information type, including but not limited to: data files, email messages and chat rooms sessions.

Within the context of this objective, the implementation of the SAMSON project was directed by an architectural approach that defined how the solution should be built so as to most easily integrate with existing networks, services and processes. This approach is described in the following section.

1.2 Project Approach

This section describes the architectural guidelines around which the SAMSON TD project was structured. These guidelines, established at the beginning of the project, served two goals:

1. To direct the creation of an innovative solution that reflects current trends towards increased security through control of data assets (smart data); and

2. To direct the development of a solution that will meet the needs of the GoC / DND user community by enhancing the security posture of operational network environments with minimal disruption of the tools, technologies and processes that are currently in use.

Each of the architectural guidelines for the SAMSON TD project is described below.

SAMSON as a Security Overlay: The SAMSON solution is to be deployed into operational network environments as a security overlay. An overlay, in this context, means three things:

1. SAMSON is a solution that integrates with or leverages security capabilities that are pre-existing in the environment. As a result, SAMSON does not mandate the deployment of new security services or require alteration to existing processes related to those services. For example, SAMSON requires access to user security attribute information (e.g. clearance, COI membership) in order to make access control decisions. SAMSON is, however, able to utilize existing IDM solutions that are currently in use in the target deployment environment. While SAMSON can supply any needed services if they are not pre-exiting in the environment, the SAMSON architecture allows organizations to continue to get beneficial return on their existing investment in security tools by leveraging, rather than replacing those capabilities.

2. The SAMSON architecture places policy enforcement at a point in between the user's workstation and the back-end data services. SAMSON natively understands the transport and data protocols of all supported applications. As a result, SAMSON does not require changes to the software suite currently in use at the workstation or the data services that are deployed to the operations center. In its role as a transparent data intercept, the SAMSON security overlay does not alter the user's experience at the workstation.

3. As an overlay, the manner by which information assets are protected can be modified and updated to reflect the current needed state of information assurance. For example, in cases where greater auditing is required, the SAMSON component that is responsible for creating a tamperproof audit trail can be amended to collect more detailed information about a specific set of transactions. Similarly, if new security services are needed (e.g. dirty word checking, digital watermarking) these

capabilities can be integrated into operational processes via the overlay without the need to disrupt ongoing service.

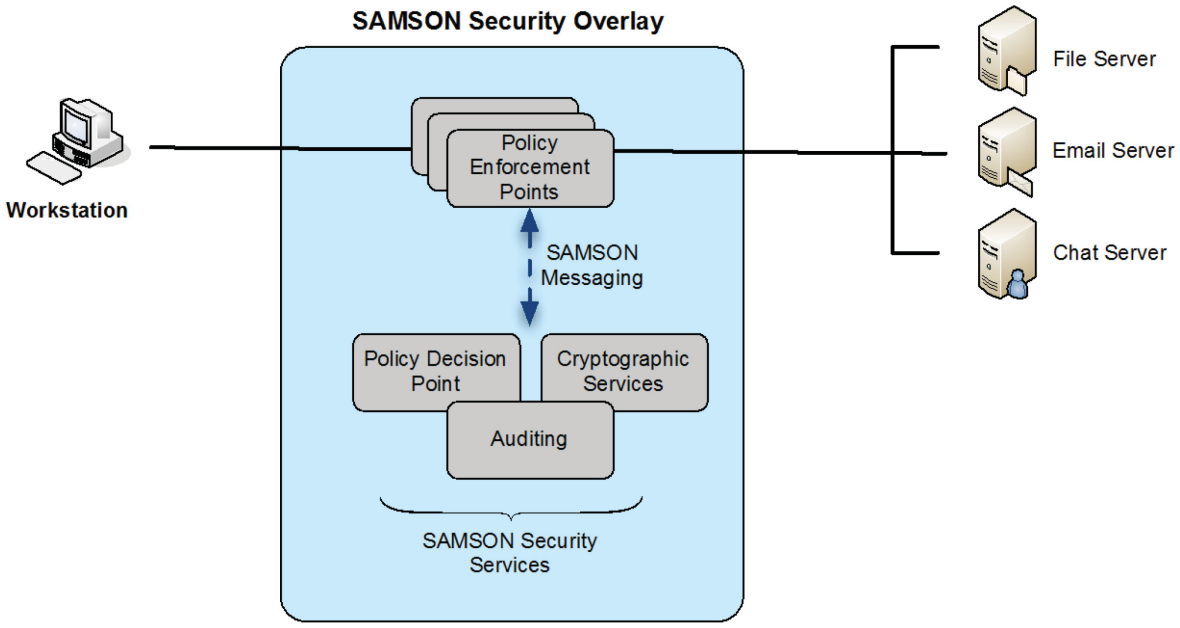

Figure 3: The SAMSON Security Overlay

As seen in the above diagram, the SAMSON architecture is overlaid on top of existing application network architectures; inserting SAMSON information protection onto existing data transmission paths. From this location, the SAMSON architecture is able to enforce information assurance through access control, information protection and auditing services.

SAMSON as a Service Oriented Architecture: The SAMSON architecture is to be implemented as a Service Oriented Architecture (SOA) where all security requirements are met by independent services that are accessible through open, well-defined interfaces.

Conceptually, the SAMSON security services act as security gateways with the ability to route an internally generated SAMSON security service information request to the actual external service or process that will handle the request. SAMSON is, therefore, not tied to any specific vendor solution and can replace any vendor solution with another product that provides similar capabilities. In this way, all SAMSON security interfaces conform to a common set of design goals, including:

- They can be made to work with any external vendor solution, product or implementation;
- They can be extended to include any SAMSON-specific capabilities that are not reflected in the chosen standard;
- They are appropriately secured in order to ensure the confidentiality and integrity of these information exchanges; and

- New security services can be added to the architecture without the need to redesign or redeploy the entire security overlay.

SAMSON as a Certified and Accredited (C&A) solution: The process of creating the data-centric security solution must include the creation of documentation artefacts that will support a C&A process. By taking the SAMSON solution through the C&A process, it is expected that there will be a clear path to the deployment of SAMSON into operational environments.

In summary, the data-centric solution created through the SAMSON TD project must:

- Be transparent so as not to disrupt existing data applications or operational practices;
- Be modular to allow existing security services to be leveraged through the SAMSON architecture;
- Be extensible to allow to new services to be deployed and used within the SAMSON information protection methodology; and
- Be trusted through a proven architecture that is in accordance with C&A security catalogues and an appropriate protection profiles.

The following section provides a more detailed description of the design strategy that allowed for the creation of a SAMSON solution that is compliant with these guidelines.

2.0 SAMSON Design Strategy

The SAMSON architecture achieves its information protection requirements through the use of three core architectural components:

1. **A Secure Messaging Service Bus (SMSB)**: The ability for SAMSON components to exchange data in a manner that is secure, protocol agnostic, and reliable.

2. **Security Services Gateways**: The ability to bridge between the SAMSON security architecture and the back end (non-SAMSON) applications that provide the needed security functionality. These services include authorization for adherence to policy, cryptographic services for protection of data and audit for the creation of a trusted chain of evidence.

3. **Policy Enforcement Points (PEPs)**: The ability to link external application and security services to the SAMSON overlay in a manner that adheres to the SAMSON security protection principles.

These components can be seen in their proper context in Figure 4: SAMSON Core Components.

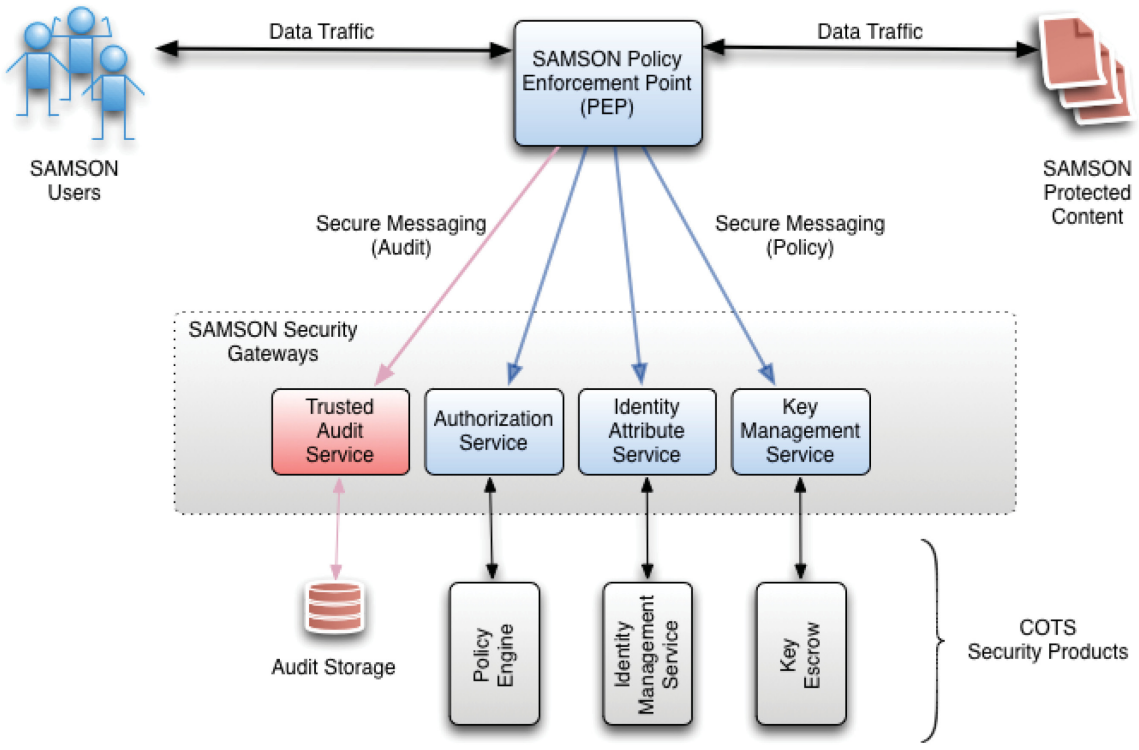

Figure 4: SAMSON Core Components

As an example of this architecture in operation, the SAMSON Authorization Service provides decisions in response to access requests on the part of the user. This policy decision is made by leveraging a Policy Engine to enforce the unified security policy for the environment. When a PEP is called upon to disclose data to a SAMSON user, the PEP formulates a policy request consisting of:

- The requesting user's identity,
- The security attributes on the data object being requested and
- The action that is to be performed on the data.

This policy request is sent to the Authorization Service that, in turn, reframes the policy request for the back end policy engine. A policy decision is returned, through the Authorization Service to the PEP for enforcement.

As an SOA, each SAMSON component is able to call upon other SAMSON services. For example, the Authorization Service can request a user's security attributes (group membership, clearance) through the Identity Attribute Service. Similarly, the PEP can submit an audit record to the Trusted Audit Service to create an immutable record of the policy decision that was enforced. The breadth of security services that are made available through the SAMSON SOA provides a complete set of information protection services needed to create a data-centric security solution. A description of the services delivered as part of the SAMSON project is described in section 2.2:Security Service Gateways.

2.1 SAMSON Messaging Infrastructure

Within the SAMSON infrastructure, any entity that communicates, either to request or provide security services, does so using industry-recognized protocols. With the responsibility to ensure robust, secure and trusted delivery of security messages between SAMSON components, the delivery mechanism is a critical core of the SAMSON architecture.

The SAMSON architecture uses the eXtensible Messaging and Presence Protocol (XMPP) as the delivery mechanism for the exchange of SAMSON security messages. XMPP is designed to support low-overhead, message-based communication over persistent sessions and is suitable for delivering encapsulated messages to authenticated entities in a secure and robust manner. XMPP is designed to be agnostic about the message payload, allowing it to carry any SAMSON message content as necessary to support the SAMSON security services. Additionally, XMPP provides the following capabilities, which are necessary to achieve trusted message delivery.

- **Authentication**: Prior to joining the messaging infrastructure, a component must be authenticated using the chosen authentication service. XMPP authenticates its participants using Simple Authentication and Security Layer (SASL). SASL supports strong authentication using public-key based credentials.

- **Persistent Communications**: XMPP is a connected synchronous communications protocol and is, therefore, resistant to man-in-the-middle or hijacking attacks.
- **Low Overhead**: XMPP is designed to handle XML messages with very low overhead that will allow SAMSON to scale to large deployments.

- **Presence Detection**: XMPP facilitates resource discovery since presence detection is built into the protocol. Resource discovery is essential to making the system scalable and reliable once operational.

- **Encryption**: XMPP sessions can be encrypted with Transport Layer Security (TLS). Encrypted sessions will not only protect the delivery details, but the message content as well. The TLS session between the XMPP clients and servers will be established using single key pair credentials issued for this specific purpose. The XMPP server will perform Certificate Revocation List (CRL) checking as part of the client authentication process.

XMPP's inherent capability for establishing presence makes it an ideal choice for supporting an SOA as it natively provides identification, monitoring, and delivery of information services within an open standard format. XMPP is complementary to the SAMSON design philosophy as it is an open standard and is XML based. Open XML standards also make it possible to extend existing XMPP capabilities to include features that are needed by SAMSON in an operational context.

While the SAMSON architecture makes no requirement on where message traffic is hosted, security best practices including: data isolation, network zoning and load balancing would recommend a deployment of SAMSON traffic along the following physically or logically segregated networks. The use of multiple Secure Messaging Service Bus (SMSBs), that is, multiple messaging domains, can be used to separate SAMSON traffic in the following manner:

1. Security Policy SMSB: An XMPP domain that supports the exchange of security information traffic such as attributes, policy requests and decisions and cryptographic key information between SAMSON components.

2. Audit SMSB: An XMPP domain that supports the delivery of audit messages between SAMSON components and the Trusted Audit Service.

While the use of separate messaging domains provides a logical separation of traffic, these domains can also benefit from physical separation. Specifically, a SAMSON deployment can attain a strongly defensible security posture through the use of separate physical networks for:

1. Management Traffic: The connection used by system administrators to manage SAMSON components.

2. Data Traffic: The network over which users connect to gain access to data services. This is the pre-existing operational network in a target deployment.

3. Security Traffic: The network that hosts the Security Policy SMSB.

4. Audit Traffic: The network that hosts the Audit SMSB.

It should be observed that an additional benefit of using separate networks for SAMSON security messages is that a SAMSON security overlay does not incur a large increase in the amount of traffic on the existing data or operations network.

2.2 Security Service Gateways

Security Service Gateways (SSGs) are participants on the SAMSON SOA that respond to security messages from other SAMSON components. These components also extend an interface outside of the SAMSON infrastructure, leveraging non-SAMSON security elements to deliver SAMSON security messages and services.

The Identity Attribute Service (IAS) can serve as a typical example of a SAMSON SSG. When SAMSON components, such as the Authorization Service, need to retrieve a user's security attributes, they do so by sending an identity attribute request via the SAMSON SOA to the IAS. The IAS itself maintains access to the back end repository where user's identity information is stored.

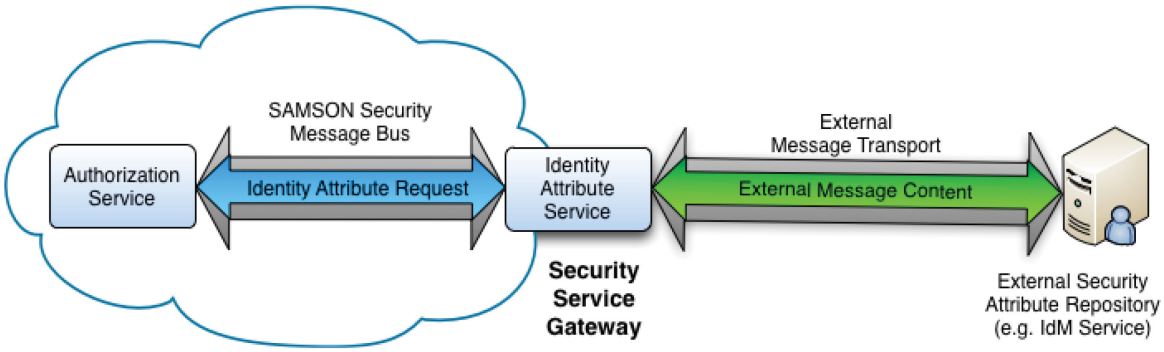

Figure 5: A Security Service Gateway (SSG)

In this way, we see that the SSG serves three key roles:

1. It *bridges* between the SAMSON SOA to extend access to security data services in the environment;

2. It *translates* protocols so that the SAMSON-based identity attribute request is reframed in the native language supported by the back end service; and

3. It *abstracts* the security service from other SAMSON components; it is possible to replace the back end identity repository with an equivalent solution without any disruption to other SAMSON services.

Six SSGs were delivered with the SAMSON TD in support of the project's goal of creating a data-centric security solution.

Identity Attribute Service: As previously described, the IAS supplies user security attribute information to other SAMSON components. Identity attribute requests are expressed using the SPML and DSML protocols. The SAMSON TD was demonstrated using two different identity repository backend:

1. The SunOne Identity Management Suite where identity information was retrieved via web application services hosted by the Sun solution; and

2. An OpenLDAP directory service.

Authorization Service: This SSG supplied policy decisions in response to XACML formatted policy requests. All PEPs in the deployed environment leveraged this PEP when actions against SAMSON protected data assets were requested. The SAMSON TD was testing with two different policy engines:

1. A custom XACML-based element matching engine using a MySQL database for hosting the security policy.

2. A custom logic-programming based solution using the XSB Prolog engine to made decisions based on predicate calculus.

Security Label Service: This SSG was responsible for extracting and verifying the security labels on file objects. Each supported data type (files, email, chat room sessions, databases) requires a different approach to linking a security label to the data object. For the SAMSON TD, the document labelling solution at the workstation is the Titus suite of classification tools (Titus Document Classification Plug-in for Office and Titus Messaging Classification Plug-in for Outlook). The deployed SLS for the SAMSON TD interprets Titus-based security labels to allow the security attributes on a labelled file to be extracted and used in policy decisions. The SLS also creates its own security label for file objects that have been protected by SAMSON. There are, therefore, external labels on data objects that are created by endpoint labelling solutions and internal labels that are placed on data objects by the SLS when an object has been protected by SAMSON.

Internal labels include cryptographic bindings to ensure that the label and content have not been altered while stored at the SAMSON protected service. The SLS created during the SAMSON TD project also included the ability to validate the label against the content; ensuring that the label is correct for the information contained in the data object. While the SLS validation routines were modular, allowing multiple validation routines to be employed against the data object, a single naïve Bayesian filter was provided for the TD project to demonstrate the validation capability.

Cryptographic Transformation Service: This SSG performs encryption and decryption operations on data in response to requests from the SAMSON PEPs. When SAMSON protects a data object, that is, when it resides on a data service that is accessed through a SAMSON PEP, that object is stored in an encrypted form. Should a malicious user attempt to access the data directly (i.e. not through the PEP), the resulting data would be encrypted with no information content disclosed. It is only by accessing data via a PEP that the cryptographic routines would be applied to decrypt the data. The PEP will, however, require a valid policy decision to allow this disclosure to take place and an audit record of the transaction would be made. For the SAMSON TD, 3rd party FIPS certified software crypto modules from RSA and Green Hills were leveraged by the CTS to perform the cryptographic transformations.

Key Management Service: This SSG supplies keys to the CTS. As part of the SAMSON trust model, each data object is encrypted with its own unique symmetric key. As a result, should SAMSON protected information be ex-filtered, each encrypted object would have to be individually decrypted using brute-force or similar cryptanalysis attacks. The keys themselves are accessed by the KMS from a key escrow service in the operational environment. For the SAMSON TD, two key escrow systems were demonstrated:

- StrongAuth SKLES; a 3rd party key escrow appliance; and
- A custom database-based key escrow system created for the SAMSON TD.

The external label that is placed on file objects that have been protected by SAMSON not only include security attributes for that file, but also include a token identifier that can be used to retrieve the key that was used to protect the file. When a SAMSON component presents a token to the KMS, the associated key is retrieved from the escrow and returned.

Trusted Audit Service: The Trusted Audit Service (TAS) is the core service for maintaining and demonstrating the integrity of SAMSON information protection processes. The TAS supports the integrity of the SAMSON trust model through the creation of audit records that are linked via a chain-of-custody to ensure tampering has not occurred within the audit trail. The audit records, protected and stored though the TAS, keep a transactional history of the policy decisions and access control enforcement across all SAMSON protected resources. Since all policy enforcement activities are recorded within the TAS and the integrity of the TAS can be demonstrated, the TAS can be used to track information access requests and the rationale for why information was disclosed to SAMSON users. The TAS also provides extensions out to SEIM solution, allowing alerts and notification messages to be raised in real-time in response to suspicious behaviour detected within the SAMSON operational processes.

In summary, each of the SSGs support one of the core capabilities within a data-centric security model:

- Access Management ensures that information is only disclosed to those users that have a policy right to perform actions against the data:

  - Identity Attribute Service;
  - Security Label Service;
  - Authorization Service;

- Information Protection ensures that information is stored in a manner where is cannot be disclosed through malicious means or ex-filtration:

  - Cryptographic Transformation Service;
  - Key Management Service; and

- Auditing ensure that there is accountability and integrity in the trust model for data protection:

  - Trusted Audit Service.

2.3 Policy Enforcement Points

Whereas security service gateways bridge and translate protocols between SAMSON and external security services, PEPs operate on data requests by ensuring the transactions adhere to the security policy, transforming the data as it is delivered and auditing the actions that have been performed against the data request. Since the fundamental goal of SAMSON is to provide information protection for data in a network environment with a pre-existing set of deployed applications, SAMSON must be able to operate on the information request formats that are currently used by those applications. In the SAMSON architecture this is achieved with application proxies that intercept and apply SAMSON information protection logic. This logic is the order in which SAMSON services must be leveraged to adequately protect the information, such as: authorization services, cryptographic services, audit services. This logic will depend on the type of data being protected and the operation on the data is being requested.

A SAMSON application proxy is placed between an end user workstation and the information resources that user is attempting to access. Note that when accessing data through a SAMSON protected resource, the user still retains use of the existing applications and the user's experience, in terms of the application activities and processes that are performed, remains unchanged. What defines a user as a SAMSON user is the fact that a set of identity attributes has been assigned to that user. These attributes are used as part of the policy checks that determine what actions the user can perform against information assets.

In Figure 6: PEP Data Mediation Process, a user is requesting information from SAMSON protected data store. The data requests go through an application PEP that intercepts the request and submits a policy decision request to the Authorization service. This data mediation process example services as an example for how SAMSON information protections works for any data type.

1. A user request a file object through the File Sharing PEP.
2. The PEP intercepts the request and calls upon the SLS to extract the security label on the requested file
3. The PEP submits a policy request to the Authorization Service, specifying the user's identity, the security attributes on the file and the action to be performed on the data
4. If the Policy decision is to allow the transaction, the PEP calls upon the CTS to decrypt the file for the user obtaining the symmetric key for the asset from the KMS. The file is then released to the user.
5. An audit record of the transaction is submitted to the Trusted Audit Store

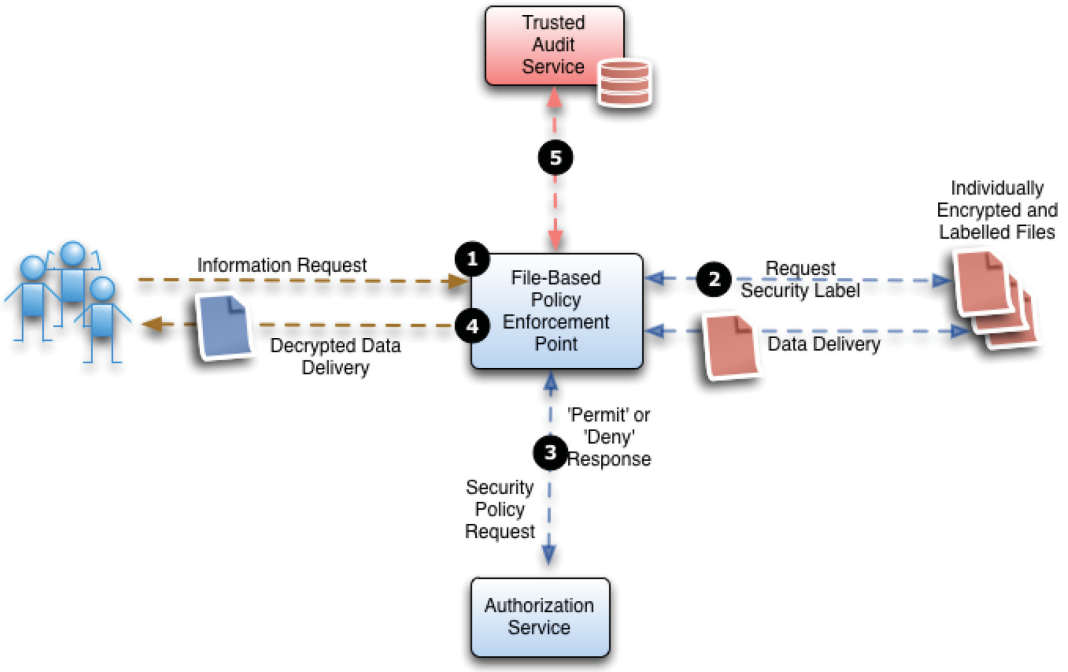

Figure 6: PEP Data Mediation Process

The following sections describe the PEPs supported by the SAMSON TD

File Sharing PEP: As described previously, the File Sharing PEP provides data centric security for data files. The PEP is placed between the user's workstation and the back end file server that hosts the SAMSON protected files. Any actions on data files are subject to SAMSON information protection logic. For example, when listing a directory, the PEP submits the user's identity and the security attributes for each file to the Authorization Service to determine if the user has a policy right to see that file and will scrub unauthorized files from the list returned to the user. As a result, the same directory listing run by two users with different policy rights will return a different list of files.

The listing, retrieval from and storage to a SAMSON protected file share conforms to the standard practices of SAMSON information assurance:

- *Access management*: based on the user's attributes and the attributes in the objects security label, does the user have the policy right to perform the action
- *Information protection*: If a file is to be stored, a new key must be generated and the file must be encrypted and re-labeled so that it is protected while residing on the SAMSON protected file share. If a file is to be retrieved, the symmetric key is retrieved from the escrow and used to decrypt the file for delivery to the user.
- *Audit*: A transaction of each policy-enforced action is submitted to the Trusted Audit Service.

The File Sharing PEP has been deployed in operational environments and used to protect:

- Microsoft File Shares hosted through MS File and Print Services; and
- Linux-based Samba File Shares.

Email PEP: The protection of email messages includes an additional challenge since policy checks must be performed not only in the email message body but also all file-based attachments that are included as part of the message. Attached files are assumed to be Titus labelled document and the message body is similarly a labelled object using the Titus Message labelling product. The SAMSON TD deployment uses an email intercept strategy based on SMTP and POP3 protocols. When a message is sent, the PEP intercepts the message, extracts the label on the message as well as all message attachments.

The sending of email messages conforms to the standard practices of SAMSON information assurance:

- *Access management*: On send, policy checks are made for the message body and each attached file. These policy checks are made for each user on the recipient list. If any recipient does not have the policy right to receive any message element, the message is rejected and returned to the sender
- *Information protection*: If an email message is to be sent, the entire MIME-encoded message is encrypted (body and attachments) using a unique symmetric key. The encrypted version of this message is stored in each recipient mailbox for retrieval
- *Audit*: A transaction of each policy-enforced action is submitted to the Trusted Audit Service.

The receiving of email messages conforms to the standard practices of SAMSON information assurance:

- *Access management*: On receive, a policy check is made to ensure that the recipient has the policy right to receive this message. If the policy check is denied, the message is not delivered, but a notification message is sent to the originating sender.
- *Information protection*: If an email message is to be delivered to the recipient, it is decrypted using the unique key for that message.
- *Audit*: A transaction of this policy-enforced action is submitted to the Trusted Audit Service.

The Email PEP has been deployed in operational environments and used to protect:

- Microsoft Exchange Services 2003/2007/2010
- Linux-based Postfix services

Instant Message PEP: SAMSON protects persistent chat room sessions. Each SAMSON chat room is labelled with security attributes and the ability to join/enter that room is subject to a policy check. Only once a user has entered a chat room can chat room messages be sent and received. Each chat room is defined with a default caveat; however, individual messages within that room can be marked up, that is, labelled with a different caveat. Individually labelled messages are subject to policy checks on send and receive. As a result, it is possible to have sub-communities within a larger community inside a single chat session. For example, in a CANUS chat room (available to users of Canadian and American nationality) is it possible for a Canadian user to send a marked up message that will only be delivered to Canadian users. American users, while still participating in the chat session, will not be aware of the segregated messages that are intended for the Canadian community only.

Messages are stored in encrypted form at the chat server. A separate key is used for each chat room / community combination. As a result a separate key is used for sub-community messages inside the broader chat room.

The receiving of IM chat room messages conforms to the standard practices of SAMSON information assurance:

- *Access management*: Users must have the policy right to join a SAMSON protected chat room given the security attributes of that room. Users must also have the policy right to generate and receive messages that are intended for a sub-community within that room.
- *Information protection*: All messages are encrypted on send and decrypted on delivery to the user's IM client
- *Audit*: A transaction of all policy-enforced actions are submitted to the Trusted Audit Service.

The SAMSON TD was deployed to operational environment where unmodified IM clients (Transverse) were used to connect, via the IM PEP, to un modified OpenFire IM servers.

Web Services PEP: The SAMSON TD project produced two types of PEP capable of providing protection for web resources. The first version of a Web PEP gated access to web resources by controlling the flow of web sessions through a proxy. Users connect to the web proxy and, subject to policy access decision, were routed through to the back end web service. The connection to the proxy was secured using a TLS protected link where the certificate used to protect then link included an attribute containing the caveat for the asset being protected. When a user connects and authenticates to the Web PEP, the user's identity and the caveat for the web session are used in a policy request.

An alternate Web PEP implementation operates directly on web data objects. In this scenario web objects retrieved from a database are include security metadata in the form of security labels. Similarly to how the File PEP reduces file directory listings, the Web PEP removes data elements from a data query response that the user does not have a policy right to access. The SAMSON TD demonstrated web session information protection using the Coalition Shared Database:

> *Standardized data dissemination is the key in achieving interoperability. A Coalition Shared Data (CSD) server, which is based on STANAG 4559 is the core of that architecture and enables the dissemination and storage of data from heterogeneous sensors from different nations, as well as tasking information and sensor data exploitation results.*

The CSD includes, as part of its schema, security metadata that includes caveat information. Web sessions, traversing the Web PEP, query the CSD for image asset records. The Web PEP uses this metadata to formulate the policy access requests that will determine which records the user has a policy right to see. Detailed CONOP information relating to the use of the CSD can be obtained through the NATO publication "Interoperable Sharing of Data with the Coalition Shared Data (CSD) Server"[2].

[2] http://ftp.rta.nato.int/public/pubfulltext/rto/mp/rto-mp-ist-086/mp-ist-086-07.pdf

3.0 Results

The results of the SAMSON TD project can be described in terms of the architectural implementations that were deployed, the testing that was performed against the architecture and the results of the certification and accreditation actives performed against the architectural baseline.

3.1 SAMSON Deployments

The SAMSON TD was successfully deployed to 4 operational exercises. Each exercise is described below.

Empire Challenge 2010: This military engineering exercise took place during the summer of 2010. EC2010 was a coalition deployment of emerging military sensor and information management tools. The centres to which SAMSON was deployed included:

- The DND Complex at the Louis St Laurent Building, Gatineau; and
- The military proving grounds at Fort Huachuca, Arizona

A separate SAMSON deployment was installed at both locations. These IT environments were classified SECRET and SAMSON was demonstrated providing caveat separation for CEO and CANUS communities. Data centric access control for file sharing, email and IM session was successfully demonstrated at both locations.

Empire Challenge 2011: SAMSON was invited to return to the subsequent EC military exercise. The SAMSON TD project extended the capability of the solution by demonstration information protection through use of per-asset symmetric key encryption.

Coalition War fighter Interoperability Demonstration: SAMSON participated in the 2011 CWID exercise where SAMSON was actively involved in the execution of military scenarios. SAMSON was deployed to the SECRET facility at the DRDC Canadian Forces Electronic Warfare Centre. For these exercises, SAMSON was used by DND personnel working directly with the technology.

Coalition Attack Guidance Experiment II: SAMSON final operational exercise, the exercise in which the full architecture and capability set was in use was CAGE II. Again held at the CFEWC, SAMSON was an integrated partner in the exercises performed during the course of the event. Full information protection across the sphere of access management, information protection and auditing was in place for all relevant applications (file sharing, email, IM and web session protection). CAGE II was again performed by DND personnel who provided useful feedback to the SAMSON team in terms of performance, ease of use and visibility. Performance metrics and data validation were collected by the SAMSON team and used as part of the C&A evidence gathering process.

In summary, SAMSON has participated in SECRET operational environment to successfully deliver data-centric information protection. In all deployments, SAMSON was successfully deployed and integrated into the exercise environment. IN each deployment, additional capabilities were added to the SAMSON architecture in accordance with the RAD development approach to delivering on the TD project. The final instantiation of SAMSON, deployed at CAGE II, has become the reference architecture against which all testing and certification activities have taken place.

3.2 Testing Results

Although unit and system testing was part of the RAD approach for the creation of SAMSON, a formal test cycle was performed against the architectural baseline. This test cycle was performed against a SAMSON deployment to the Classified Test and Development Center (CTDC), a representation of the Consolidated SECRET Network Infrastructure (CSNI).

This formal testing that took place under this effort examined the installation, configuration, acceptance testing, performance, scalability, and stress testing on SAMSON to:

- Define the SAMSON configuration for the CTDC;
- Define the SAMSON roles specific to the CTDC environment;
- Determine the time and effort required to install and configure SAMSON from “bare” machines;
- Provide a complete listing (Software Version Control) of all SAMSON and 3rd Party software used for the CTDC installation;
- Determine the resources and level of effort required to operate and support day to day operations of the SAMSON environment;

The SAMSON application and security services were tested using end-to-end systemic test scenarios run from user workstations and unmodified user applications. Test coverage was addressed by carrying out a full set of file services, instant messaging, and email functional tests. In addition, all audit records associated with the tests were recorded and analysed for correctness and accuracy in reporting policy violations. Performance and Scalability testing carried out as part of the CTDC Trial (SD-006) indicated that a SAMSON deployment based on the CTDC baseline configuration could support a user community size and performance levels as defined below.

- File services: A user community size of 1000 users, where 400 are active in transferring a 1 MByte file every 5 minutes.
- Instant message services: A user community of 1000, where 150 are active and carrying out a chat session every 20 seconds.
- Email services: A user community size of 1000 users, where 150 are active in sending and receiving an email every 5 minutes.

3.3 Certification and Accreditation Process

SAMSON Certification and Accreditation (C&A) activities were supported by:

1. Security Testing based on a Common Criteria (CC) EAL3 methodology and
2. A Security Target developed for the SAMSON system.

The C&A objectives were designed to meet the Confidentiality, Integrity, Availability, Auditability, and Non-Repudiation, within Target Infrastructures at different classification levels (Top Secret, Secret, and Unclassified).

A review was carried out on the various enterprise and product security control catalogues, namely the ITSG-33, NIST SP800-53, and the Common Criteria, Evaluation Assurance Levels. It was determined that, using a product based approach, the most effective security assurance coverage would be obtained by using the Common Criteria at the EAL3 level. A SAMSON Security Target was developed based on the NSA Labeled Security Protection Profile (version 1b). The Samson Security Target provided a listing of Security Functional Requirements for Audit, User Data Protection, Identification and Authentication, Security Management, Protection of the TOE, and Cryptographic Support for the SAMSON product.

Security testing was carried out by developing the test cases for each requirement and documenting the results in terms of *Met*, *Partially Met*, or *Not Met.* Test coverage was achieved by carrying out 93 security tests to address each Security Functional Requirement in the Samson EAL 3 Security Target. A Threat and Risk Assessment conducted by DRDC, and developed as part of the C&A process, addresses the Threats to the Samson system, and determines the required Safeguards to ensure a Low Residual Risk.

4.0 Future Work

During the course of the SAMSON TD project, certain aspects of the implementation were identified during development meetings, through After Action Reports (AARs), and during C&A discussions that pointed to gaps, needed functionality and general areas of improvement for the SAMSON architecture. These items are presented here as a road map for future work to extend the capability and improve the trust model for the SAMSON solution.

These work items are grouped using a risk management approach. This approach looks at each work item remaining and evaluates it for the Level of Effort (LOE) required to bring the work item to an operational level against the Operational Requirements and CF Needs (Rare, Unlikely, Possible, Likely Certain). LOE in this approach includes the research required to define the problem space and design the solution that will meet CF Needs. LOE also include the degree to which the solution will pose a technical challenge for implementation and integration within the SAMSON architecture.

Note that this list presents a set of SAMSON extensions and enhancements that are seen as the next logical step towards maturing the technology. This set is not an exhaustive list of potential improvements and it is anticipated that new priorities for SAMSON enhancements will be defined as the technology is adopted.

1. Cross-domain Data Exchanges: The current SAMSON implementation is meant for a single domain exclusively. An investigation into the challenges of supporting multiple domains would be a logical setup towards greater SAMSON uptake. These issues would include: cross domain security label interpretation, cross domain policy evaluation, extending the trust model to disclose of data between domains and the requirements for enhanced auditing. This enhancement will require a high degree of research and a moderate-high degree of development for implementation. The LOE for this activity is seen as **high**. (Operational requirement: **Certain**)

2. Native Exchange Support: The current SAMSON implementation intercepts email using the SMTP and POP3 protocols. While sufficient for a demonstrator, in operational environments an intercept that can work directly on the native Microsoft Exchange MAPI/ RPC protocols is deemed to be necessary. While this is at odds with the SAMSON philosophy of working on open standards and protocols, native Exchange support is seen as essential to the acceptability of the solution. This enhancement will require a low-moderate degree of research and a moderate degree of development for implementation. The LOE for this activity is seen as **moderate**. (Operational requirement: **Certain**)

3. Encrypted File Search: Since data files are encrypted when protected by SAMSON, there is no way to search through file content. A mean by which searches can be performed against files while in no way compromising the confidentiality or integrity of the data is needed. The SAMSON team has performed some initial investigation into encrypted search indices; a more in-depth examination of this challenge is needed. This enhancement will require a moderate degree of research and a moderate-high degree of development for

implementation. The LOE for this activity is seen as **moderate-high**. (Operational requirement: **Likely**)

4. Data-Centric Database Protection: Data-centric information protection of databases has been identified as being of significant interest to DND/CF and the GoC in general. Some preliminary work with CryptDB (an MIT research initiative) identified a path to a SAMSON solution for database support. By leveraging advanced cryptographic concepts such as homomorphic encryption, SAMSON compliant PEPs for databases could be created, but more research would be required to progress this solution. This enhancement will require a moderate degree of research and a high degree of development for implementation. The LOE for this activity is seen as **moderate-high**. (Operational requirement: **Certain**)

5. Logic-Based Policy Engine: It is the belief of the development team that a policy engine based on a predicate calculus is a better approach to creating PDPs. A simple Prolog-based PDP was created and demonstrated during the project but a complete solution that is fully-compliant with XACML expressions would be an improvement over the existing PDP implementation. Human readable policies can be more easily express using a logic-based policy engines leading to more accurate and flexible policy expression. As a result, however, this effort will also require the development or integration of a policy editor that is able to create logic-based policy expressions. This enhancement will require a moderate degree of research and a moderate degree of development for implementation. The LOE for this activity is seen as **moderate**. (Operational requirement: **Possible**)

6. Improved Labelling of Data Objects: The approach taken to label certain data objects needs to be re-visited to ensure that the labels that are created confirm to a labelling standard, can be bound to the asset and support the trust model. Specifically, a better method to label file share directories and chat room sessions is required. When examined in a general sense across all data types that will require improved labelling solutions, this enhancement will require a moderate degree of research and a low-moderate degree of development for implementation. The LOE for this activity is seen as **low-moderate**. (Operational requirement: **Likely**)

7. Policy Enforcement Extensions: While the XACML standard includes the ability to include environmental conditions in policy expressions and policy requests, the existing deployment does not leverage this capability. The existing PEPs can be enhanced to include: time of day and location information as part of the policy decision process supporting the creation and enforcement of more sophisticated security policies. This enhancement will require a moderate degree of research and a low-moderate degree of development for implementation. The LOE for this activity is seen as **low-moderate**. (Operational requirement: **Unlikely**)

8. Improvements to the Intercept technologies: Both the file and IM PEP exist as modifications to existing data intercept technologies. While useful for a demonstration, standalone PEP that can more easily be supported and deployed would add to the robustness of the SAMSON solution. This enhancement will require a low degree of research and a low-moderate degree of development for implementation. The LOE for this activity is seen as **low-moderate**. (Operational requirement: **Possible**)

9. More SAMSON services: In addition to the services defined above, there has been an identification of additional security services that could be of benefit in a SAMSON deployment, including: PKI-based encryption of data assets when disclosed to users and digital watermarking of disclosed data assets. Similar to enhanced labelling, this enhancement will have differing levels of effort depending on the type of service to be added to the SAMSON architecture. Assuming that the nature of the services to be deployed is well defined, this enhancement will require a low-moderate degree of research and a low-moderate degree of development for implementation. The LOE for this activity is seen as **low-moderate**. (Operational requirement: **Likely**)

The amount of work remaining defined above indicates the remaining research and development components required for a fully operational deployment in the near term. Near term is defined as being of a Technology Demonstration Project size and complexity.

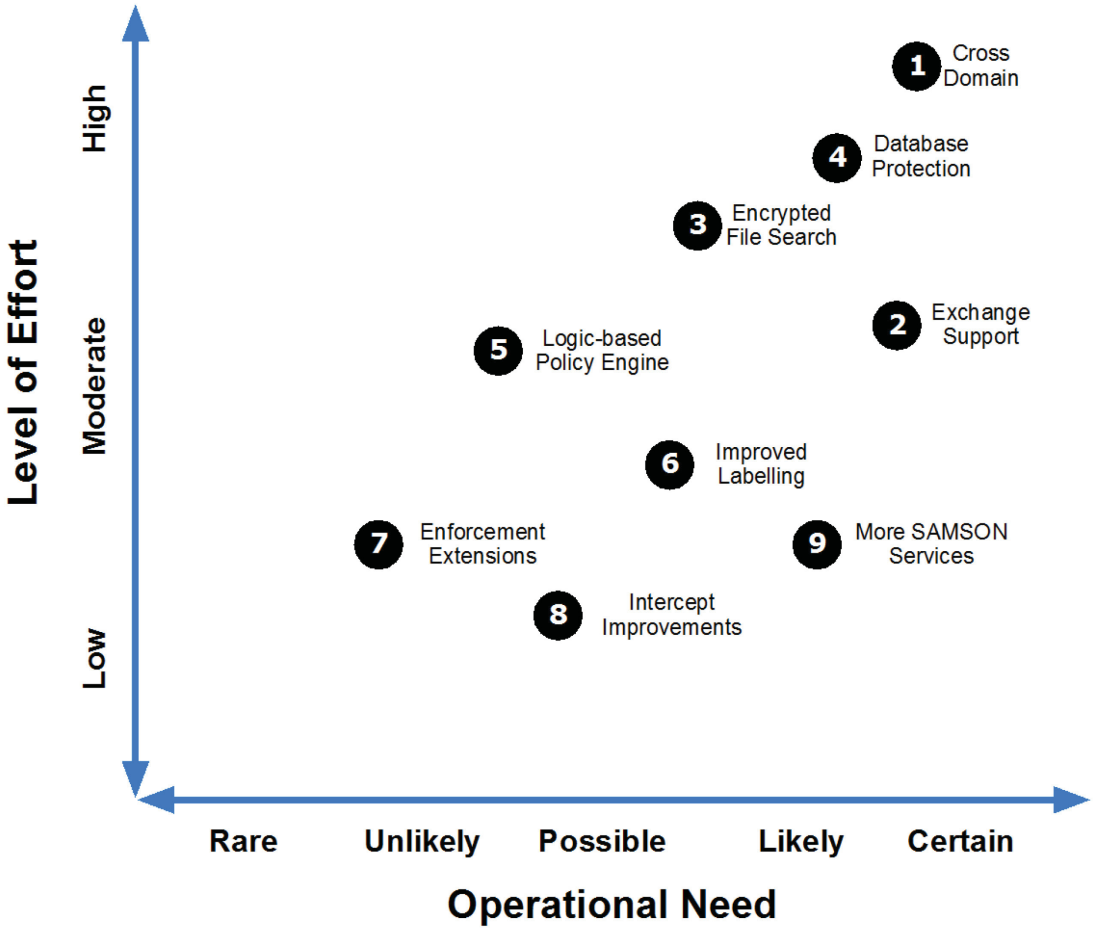

Figure 7: Future SAMSON Work

4.1 Action Plan

From the information presented in Figure 7: Future SAMSON Work, a future work action plan and response matrix can be created.

ACTION RESPONSE	RESPONSE DESCRIPTION
Long Term Research	There is a conscious decision, based on documented acceptance rationale, that this work package cannot be accomplished in a reasonable time frame (4-6yrs). A longer term R&D effort must be initiated.
Near Term Research	This work package is sufficiently mature that a Technology Demonstration Project type of effort is required to mature the work package and demonstrate the capability..
Development / Implementation Effort	This work package has been partially demonstrated, but portions of are insufficiently developed to transfer to industry. This work package requires a specific focused Development effort that will reduce the risk exposure, by reducing either the overall R&D remaining or mitigate the operational need.
Transfer	Transfer the work package to a third party. This party can be a prime contractor, sub-contractor, existing DND expertise or other government department.
Watch	Conscious decision, based on documented acceptance rationale, to accept the associated level of work can be accomplished by industry, without engaging in any active R&D efforts to control it.

Table 1: Action Responses

1. Cross-domain Data Exchanges: **Near Term Research**

   The current SAMSON implementation has demonstrated a primitive cross-domain capability in the CAGE exercise. From this work the issues have been identified but not addressed. This work requires significant R&D Scientific Authority input and incremental capability demonstration in order to address the work package.

2. Native Exchange Support: **Transfer**

   The current SAMSON implementation has demonstrated MAPI / PRC protocols in a lab setting for specific demonstrations, this is insufficient for an operational deployment. This effort will be required by National Defence, but requires minimal R&D support from the Scientific Authority.

3. Encrypted File Search: **Develop**

   Specific DND operational communities require this file search capability. This is a moderate to high amount of R&D effort, and requires a dedicated R&D task to accomplish this work. This work requires minimal amount of Scientific Authority support and moderate contractor support.

4. Data-Centric Database Protection: **Transfer**

   This work has been demonstrated on a test database within a lab environment. The basic R&D concepts have been demonstrated, with a large amount of development work remaining.

5. Logic-Based Policy Engine: **Watch**

   This work is nearly a commercial capability, and is being worked on by various commercial entities. This may need modification to work within a military environment, but can be transferred to a capable third party for implementation.

6. Improved Labelling of Data Objects: **Develop**

   This work will depend on the operational needs assessment for a deployed SAMSON capability. Some R&D Scientific Authority work is required as new labels and label standards have a broad-spectrum impact across all of the SAMSON infrastructure.

7. Policy Enforcement Extensions: **Watch**

   This work will depend on the operational need for a deployed solution. This requires minimal Scientific Authority expertise and can be transferred to a capable third party to perform the work

8. Improvements to the Intercept technologies: **Watch**

   This work will depend on the operational need for a deployed solution. This requires minimal Scientific Authority expertise and can be transferred to a capable third party to perform the work.

9. More SAMSON services: **Develop**

   This work will depend on the operational needs assessment for a deployed SAMSON capability. Some R&D Scientific Authority work is required as new services will require slightly modified SAMSON capability to support.

Future Work	Policy-based Logic Engine **(5)** Policy Extensions **(7)** Improved Intercepts **(8)**	Native Exchange Support **(2)** SAMSON Database Protection **(4)**	Encrypted File Search **(3)** Improved Data Labeling **(6)** More SAMSON Services **(9)**	Cross Domain Data Exchange **(1)**	
	Watch	Transfer	Development / Implement	Near Term Research	Long Term Research

Table 2: Future Work Action Plan

Table 2: Future Work Action Plan presents the prioritized R&D work packages. As can be seen, there are no remaining R&D components that are considered long term R&D. The SAMSON team has matured the majority of risks and work packages over the last 4 years of development and demonstration. However, there are short to mid term investment priorities that can serve to expand upon the work done during the demonstrator phase of the project and will position the solution to meet immediate operational needs. These priorities and the level of GoC-sponsored Scientific Authority involvement required are identified in the next section.

4.2 Research Effort Priorities

This section summarizes the defined areas of future work for the continued maturing of the SAMSON data-centric information protection model in the context of research priorities. The areas of research investment are presented in terms of high, medium, low and very low priority:

Research Priority	Rationale
HIGH	High priority research items are defined as having a high operational need and a moderate-to-high level of effort. Specifically, high priority research items have a significant research component that will involve *significant participation by the Scientific Authority* assigned to guide the effort.
MODERATE	Moderate priority items have a high operational need coupled with a moderate level of effort. The research commitment to the effort is low to moderate and a *moderate involvement of the Scientific Authority* is required.
LOW	Low priority items have a moderate likelihood of meeting an operational need and have a low to moderate level of effort. There is *minimal need for Scientific Authority* involvement in the effort.
VERY LOW	Very Low priority items have no anticipated operational need in the mid-level time frame. There is a low-to-moderate level of effort to implement the item and *no Scientific Authority involvement is required*.

Table 3: Research Priorities

The following diagram presents the Figure 7: Future SAMSON Work, with the research priorities overlaid on the existing

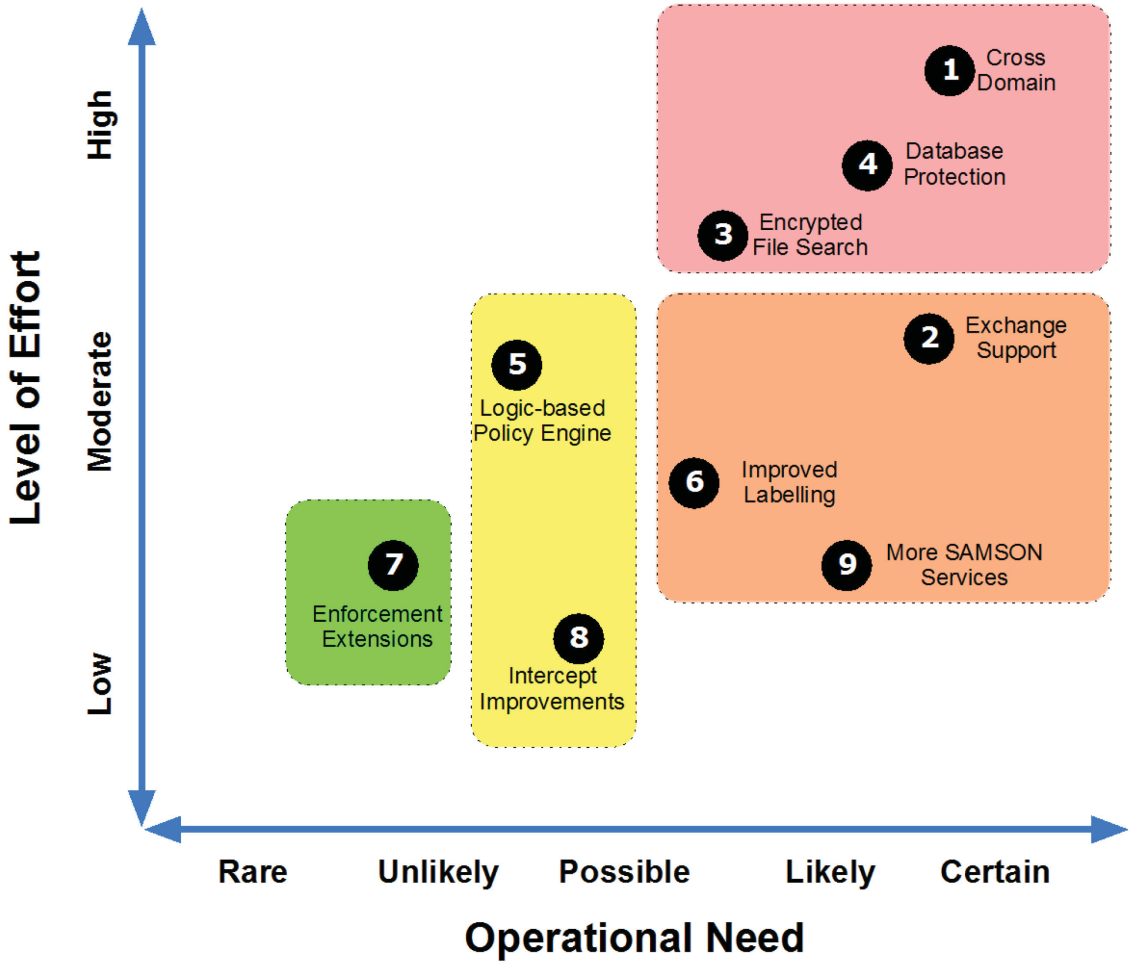

Figure 8: Prioritized Future Work Items

From this diagram, the remaining research priorities that require Scientific Authority support are clear

1. Cross Domain Data Exchange (1);
2. Data-Centric Database Protection (4); and
3. Encrypted File Search (3).

5.0 Conclusions

In conclusion, the SAMSON TD project has been successful from a number of perspectives:

1. It has advanced the state of information protection to create an enterprise-wise approach to data-centric security.

2. It has demonstrated the viability of the solution in SECRET operational contexts.

3. It has been shown to adhere to security catalogues associated with protection profiles for similar solution in this data-centric security space.

4. It has demonstrated the innovative use of technologies to create a security overlay that can both integrate seamlessly with existing operational environment while providing an extensible architecture to which new security services can be added.

5. It has shown that it is possible to build a unified security architecture based on open standards and open protocols.

6. It has supplied the artefacts in support of these points in a certification and accreditation context.

The scope of the SAMSON project was deliberately contained to the protection of information in a single security domain. The results of this project have proven that a data-centric security solution can be integrated as a security overlay onto an existing IT architecture. SAMSON has been successfully demonstrated enhancing the information protection and information sharing capabilities for operational environments. In observing the development and success of the SAMSON project, demand has been generated from NATO and other coalition partners for a data-centric security solution that can bridge between coalition security domains.

To meet this demand and further the range of applicability for the SAMSON model, support for cross-domain information sharing and protection is seen as the most significant area of research for the next phase of SAMSON development. Research into this expanded capability set will include the need for: interpretation of security metadata across domains, expression and interpretation of multi domain security policies, maintaining the trust model as data is exchanged between domains and enhancements to trusted auditing.

The SAMSON TD project can service not only as a demonstration of an innovative security architecture for the next generation IT environments, but also as an example of how research projects should be structured and executed to ensure a successful result.