NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

MISUSE CASE DRIVEN DEVELOPMENT OF SECURE INFORMATION SHARING FOR COALITION ENVIRONMENT

by

Seung Soo Baek

September 2007

Thesis Advisor:	J. Bret Michael
Co-Advisor:	Duminda Wijesekera

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE				*Form Approved OMB No. 0704-0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** September 2007	**3. REPORT TYPE AND DATES COVERED** Master's Thesis		
4. TITLE AND SUBTITLE Misuse Case Driven Development of Secure Information Sharing for Coalition Environment				**5. FUNDING NUMBERS**
6. AUTHOR(S) Seung Soo Baek				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000				**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A				**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited				**12b. DISTRIBUTION CODE**
13. ABSTRACT (maximum 200 words) Sharing information among communities can result in more informed decisions being made faster. Information sharing involves the flow of unclassified and classified information, and consequently should be carefully engineered to avoid flow-based mistakes such as creating covert channels inadvertently. This thesis uses misuse cases to identify such misuses of a sharing system. We show that an appropriate distributed role-based access control model imposed upon information brokers can prevent enumerate misuse cases. We use the North Korean nuclear proliferation as a case study to elucidate our claims.				
14. SUBJECT TERMS Information Sharing, Use Case, Misuse Case, dRBAC, Access Control Policy, Need to Share, North Korea, Nuclear Weapon's Detection				**15. NUMBER OF PAGES** 82
				16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified		**20. LIMITATION OF ABSTRACT** UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

MISUSE CASE DRIVEN DEVELOPMENT OF SECURE INFORMATION SHARING FOR COALITION ENVIRONMENT

Seung Soo Baek
Captain, Army, Republic of Korea
B.S., Korea Military Academy, 2002

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
September 2007

Author: Seung Soo Baek

Approved by: Professor J. Bret Michael
Thesis Advisor

Professor Duminda Wijesekera
Co-Advisor

Professor Peter J. Denning
Chairman, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Sharing information among communities can result in more informed decisions being made faster. Information sharing involves the flow of unclassified and classified information, and consequently should be carefully engineered to avoid flow-based mistakes such as creating covert channels inadvertently. This thesis uses misuse cases to identify such misuses of a sharing system. We show that an appropriate distributed role-based access control model imposed upon information brokers can prevent enumerate misuse cases. We use the North Korean nuclear proliferation as a case study to elucidate our claims.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I would like to express my sincere appreciation to Professors J. Bret Michael and Duminda Wijekesera, my thesis advisor and co-advisor, for their patience and guidance throughout my work on this thesis.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. OVERVIEW

The purpose of this thesis is to develop secure information sharing for coalition environments. A coalition environment refers to the need to share information between two or more communities of interest. Sharing information among communities can result in better and faster decision making. After the attacks on the United States on September 11, 2001 (9/11), the U.S. and its allies began to broaden the sharing of information among government and nongovernmental organizations. To date, the U.S. government still lacks a dynamic, decentralized network for sharing and analyzing information. The sharing of information between relevant agencies at different levels of government is still dependent on multiple systems with limited or no ability to communicate with each other and is still constrained by institutional and technical barriers. Fragments of data collected by different agencies are likely to remain in different places with no way to find them and therefore no way to make sense of what is happening. Because users from several communities want to use the same information at the same place, the distribution of information can be a critical problem in a role-based access control model. In order to solve this problem, the thesis introduces the concept of distributed role-based access control (dRBAC) and the information broker (IB) as security-enhanced access control policy. This thesis explores the tenets of the information broker,[1] a highly trustworthy agent in charge of distributing information from repositories in a multilevel security (MLS) context. Moreover, the thesis introduces the relationship between local and global information brokers.

As the environment of inter-community information exchange is decentralized and increased, security has become a significant problem. Sharing information could involve controlling information flow between the entities by means of the security policy. However, security policy does not resolve all the security problems of information

[1] C. R. MacDaniel, and M. L.Tardy, "Role-Based Access Control for Coalition Partners in Maritime Domain Awareness," M.S. thesis, Naval Postgraduate School, Monterey, CA, June 2005.

sharing. Before implementing a shared information system, one must consider the security aspect of information flow within the coalition environment. Early consideration improves the developer's ability to build the right policy and requirements into the information system. This thesis introduces misuse cases for making secure-enhanced information sharing. Misuse cases help reveal the security risks of information sharing and what use cases can help to mitigate those risks.

In addition, the thesis presents a scenario involving detection of North Korean nuclear weapons. The scenario is used to illustrate the implementation of dRBAC with IBs to achieve secure information sharing.

The first step in this approach is to decompose the mechanism of information sharing. Information sharing can have several sub-functions, such as (1) identifying users, (2) checking location of information, and (3) providing requested information. Next, a unified modeling language (UML) for misuse cases for secure requirements of the system.[2] Misuse cases model undesirable behaviors of the system, such as a disgruntled system administrator who inserts a logic bomb into an organization's information system. Misuse cases allow the creation of a new model of secure information sharing with dRBAC.

Chapter I provides background information on the North Korean nuclear weapons problem, the information broker, and several theories relevant to the application such as RBAC, dRBAC, and misuse cases for decomposing and composing secure requirements. Chapter II introduces a decomposition of functions in information sharing. Chapter III covers the maintenance of the decomposition and complementary misuse cases for security requirements. Chapter IV demonstrates the approach with a realistic hypothetical scenario. Chapter V provides conclusions and possible areas for future research.

[2] G. Sindre and A. L. Opdahl, "Eliciting Security Requirements by Misuse Cases," *Proc. 37th Conf. Techniques of Object-Oriented Languages and Systems*, TOOLS Pacific 2000, 120-131.

B. BACKGROUND

This thesis addresses the challenge of providing information sharing in a coalition environment setting in which members of the coalition and the roles they play are in constant flux. Information sharing within coalition environments is needed to counter threats to national security. Terrorism, for instance, knows no borders. To obtain a common operating picture (COP) to counter threats takes cooperation among nations, in addition cooperation between the public and private sectors, and with different degrees of need to share. However, suppose that a hacker in North Korea wants to interrupt the sharing of information between coalition partners. The thesis uses misuse cases to describe the risks when coalition states want to exchange information. Although members are in an alliance, they might be unfamiliar with the opposition, as in the case of the U.S. and China, countries that do not want to show their information to each other. Neither case guarantees the fluid exchange of shared information.

The thesis builds on the concept of the information broker as a wallet of dRBAC to facilitate communication between allies. The information broker handles requests by users to access repositories of information. The user does not have a need to know what repositories are available, or which repositories supplied the answer to the user's query; that is, the information broker maintains the anonymity of the repositories. The thesis treats the dRBAC model as an access controller of the information broker. There is a variety of vulnerabilities to information flow in a multilevel secure system environment such as a hacker's interception of information flow. Therefore, consideration of security in the model is essential to protect against the vulnerabilities.

1. North Korea's Nuclear Weapon Program

The nuclear weapon problem has arisen among nations because nuclear weapons create environmental, political and social problems such as radioactive contamination and foreign policy challenges. For over ten years, North Korea has been the focus of global concern about nuclear proliferation. Recently, North Korea declared that it possesses several nuclear weapons and has successfully conducted a test explosion of a nuclear weapon. In response to this proliferation of nuclear weapons, six neighboring nations met in Beijing, China and adopted a declaration. However, North Korea has not complied with the resolution from the six-party talks. The thesis therefore assumes that the detection of a nuclear weapon is a possible scenario.

a. Detection of North Korean Nuclear Weapons

This scenario is built on a weapon detection problem involving North Korea. North Korea poses a threat to North America, Europe, and Asia in this scenario. The scenario assumes that North Korea's nuclear weapons program is detected by a military intelligence agency.

In the first step of the scenario, the Republic of Korea (ROK) detects suspicious objects in North Korea and informs the United Nations (UN) of that fact. In the second step, the members of the UN Security Council verify the fact that North Korea has a veiled nuclear weapon, based on information from repositories in ROK, China and the U.S., and makes contingency plans to solve the problem. The last step in the scenario demonstrates how to support each ally in accomplishing the operation by sharing information.

2. Need to Share Security Policy

Before 9/11, those holding classified data applied the "need to know" model when considering information sharing. Since 9/11, there has been a call for improved information sharing, which has resulted in "need to share."

"Need to know" and "need to share" are not parallel terms. "Need to know" is often used as a content-producer label to mark data assets with the assertion that the data should not be shared unless the user/consumer meets certain criteria. However, "need to share" is a concept, a precept and an objective. So many data assets are created; they need to be visible, accessible, and understandable so that they can be shared.

Sharing information can facilitate better, faster decision-making at all levels of government. It has several key features.[3]

> Shared information comes from a decentralized network. It sends and pulls information from all participants in counterterrorism efforts, from local law enforcement officers to senior policy makers. In addition, sharing information is a hybrid of technology and policy. The information sharing system uses currently available technology to share and protect the information that flows through it. When paired with clear guidelines, it could determine the collection, use and retention of information, and who should have access to information; it can both empower and constrain intelligence officers, and provide effective oversight. In addition, sharing information allows for vertical and horizontal coordination and integration. Information flows not just up the chain of command, but also horizontally, to the edges of the system. Finally, sharing information enables analysts, law enforcement agents and other experts to find others with common concerns and objectives and to meet in informal "virtual" teams to exchange information and ideas.

3. The Information Broker

There are many advantages to “need to share.” However, sharing information also produces the “information sharing paradox.” If a coalition member does not want to share everything with the others, the member cannot ask anyone any questions. Then how is a member able to find something? To solve this paradox, the information broker is introduced.

The information broker is an information management controller in the information exchange system who acts as an intermediary between the requester of the information and the data repository. The information broker provides the data and at the same time shields the source of that data from the requester. In other words, the information broker encapsulates the data or the request under its own name and thereby maintains the confidentiality of the requestor and the repository.

This is a black box approach that can satisfy the data requests and protect the source. The information broker is based on the dRBAC model. By using dRBAC, the

[3] James X. Dempsey."Moving from 'Need to Know' to 'Need to Share:' A Review of the 9-11 Commission's Recommendations." Center for Democracy & Technology. Retrieved July, 2007, Available from http://www.cdt.org/testimony/20040803dempsey.shtml.

information broker can control the distribution of information and authenticate personnel. The information broker is intended to be a highly trustworthy component of a system, responsible for dealing with a myriad of clearances, classifications, and compartments.

4. Role-based Access Control

Role-based access control (RBAC) is a means for controlling access to computer resources by associating access permissions with roles. Users are given roles to help simplify the management and enforcement of access control policies. The genesis of RBAC can be traced to the emergence of multi-user and multi-application on-line systems. The dRBAC model described here is based on the role-based access control principle. The assignment and membership principle is central to simplifying the management of permissions.

With RBAC, access decisions are based on individual users' roles in an organization. Users take on assigned roles such as the Republic of Korea's National Intelligence Service (NIS) observers and International Atomic Energy Agency (IAEA) investigators. The process of defining roles should be based on a thorough analysis of how an organization operates and should include input from a wide spectrum of users in the organization.

The case model of RBAC consists of a user, a role, a session, and permission. A user in this model is a human. A role is a job function or a duty within some organization. A NIS agent searching for some information associated with an event is an example. A session occurs when an issuer activates some subset of roles to which he or she belongs. Permission is a particular model of access granted to one or more objects in the system.

The RBAC model captures the following types of relationships: many-to-many permissions to role assignment and many-to-many users to role assignment. A function maps each session to a single user. Each session also gets mapped to a set of roles.

Hierarchies are also a part of RBAC: they structure roles to reflect an organization’s line of authority and responsibility. In addition, they form a partial order relationship, which means that they are reflexive (role inherits its own permission), transitive, and anti-symmetric. In short, a subject may have multiple simultaneous sessions with different permissions.

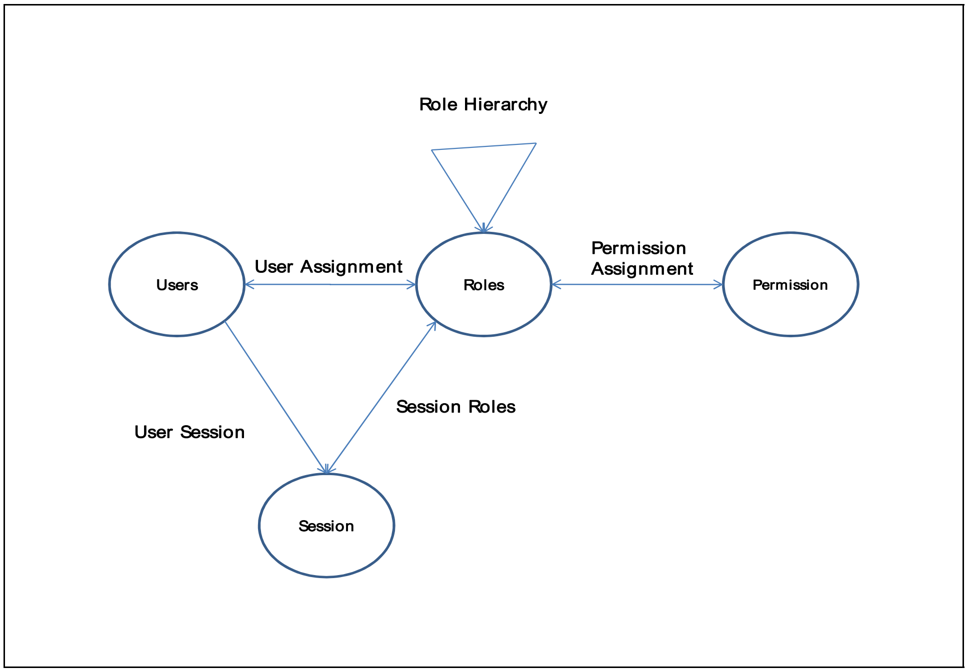

Figure 1. Hierarchical RBAC[4]

5. Distributed Role-based Access Control

dRBAC is a scalable, decentralized trust-management and access control mechanism for systems that span multiple administrative domains. The development of dRBAC was motivated by the problem of controlling access to resources in a coalition environment. A coalition environment might be commercial, in which corporations from several nations work together to achieve a common goal. The entities must cooperate to share the subset of their protected resources necessary to the coalition while at the same time protecting the resources that they do not want to share.

[4] Figure 1 is drawn from D. F. Ferraiolo, R. Sandhu, S. Gavrila, D. R. Kuhn and R. Chandramouli. "Proposed NIST standard for role-based access control. *ACM Transactions on Information and System Security*, 4(3): 2001, 224-274.

The growth of distributed systems faces a variety of challenges in a coalition environment.[5]

1. Dynamic coalition environments have to provide organizations and allies with the authorization of resources at varying levels of access to interact.
2. Established trust relationships must be monitored over their lifetime to track the status of revocable credentials.
3. Credentials that authorize a desired trust relationship must be distributed automatically to those who require them.

Traditional approaches to employing RBAC in information systems rely on a central trusted computing base administered by a single authority. However, dRBAC is distinguished from previous approaches because it combines RBAC and trust-management systems to create a system that offers both administrative ease and a decentralized scalable implementation.[6]

1. Third party delegations allow an authorized entity to delegate roles created by another entity by referring directly to the role originator's namespace.
2. Values attributed supports another control of access right when supporting varying levels of access for the same resource.
3. Continuous monitoring allows dRBAC to guarantee validity of established trust relationships over the lifetime of prolonged interactions.

These three features of dRBAC enable the construction of a trust management and access control system.

Delegations are published, validated, updated and revoked using dRBAC wallets[7] as a part of the information broker. This wallet is similar in function to PKI certificates and can store many delegations. In addition, for credential management, the wallet has

---

[5] Eric Freudenthal, Tracy Pesin, Lawrence Port, Edward Keenan, and Vijay Karamcheti. "dRBAC: Distributed Role-Based Access Control for Dynamic Coalition Environments" (TR2001-819), *Proceedings of the Twenty-second IEEE International Conference on Distributed Computing Systems (ICDCS)*, Vienna, Austria, 2002, 411-420.

[6] Freudenthal *et al.*, 412.

[7] Freudenthal *et al.*, 416.

features such as discovery tags and proof monitoring. Discovery tags include an Internet address identifying authorized wallet, wallet's name, time-to-live, and search flag. Through discovery tags, dRBAC establishes the trust relationships between wallets and/or data repositories. Another significant feature is the proof monitor. Proof monitors register delegation subscriptions with trusted wallets for each delegation in the proof. This proof monitor provides for the integration of all the proofs of authentication.

C. SECURITY REQUIREMENTS

1. Misuse Case

A misuse case is defined as a special kind of use case, describing behavior that the system/entity owner does not want to occur.[8] A misuse case has all the same properties as an ordinary use case. Given this, who handles the misuse cases? The answer is a mal-actor, a special kind of actor who initiates a misuse case. The mal-actor can be from inside or outside the system. A mal-actor can be a rogue or legitimate user. Figure 2 depicts an example of a misuse case and a mal-actor.

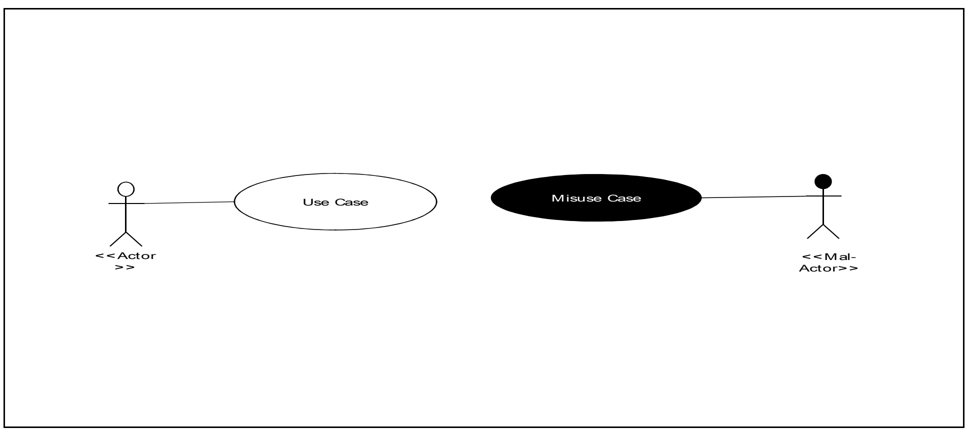

Figure 2. Example of Misuse Case and Mal-actor

[8] G. Sindre and A. L. Opdahl, "Eliciting Security Requirements by Misuse Cases," *Proc. 37th Conf. Techniqus of Object-Oriented Languages and Systems*, TOOLS Pacific 2000, 122.

2. Relation between Use Cases and Misuse Case

a. Include and Extend

In the UML diagram, there are two relationships, namely, “include” and “extend.” When a base use case is needed to do an inclusion use case, the relation between the two use cases can be defined as “include.” The “extend” relationship is used for a part of a use case that is optional system behavior.

b. Threaten, Mitigate, Prevent and Detect

In addition to the standard “include” and “extend” relations, security requirements introduce other relations. These are particularly interesting with respect to use cases involving misuse cases. They are called “mitigate,” “detect” and “prevent.” In contrast, in a relation that comes from misuse it is effective to use cases negatively. This relation is called “threaten.”

- **Threaten**: The function provided by the misuse case where the arrow originates from a mis-actor and threatens the activation of the use case that the arrow is directed towards, at least in some cases.
- **Mitigate**: The function provided by the use case that the arrow originates from that mitigates the activation of the misuse case that the arrow is directed towards, at least in some cases.
- **Prevent**: The function provided by the use case that the arrow originates from that prevents the activation of the misuse case that the arrow is directed towards, at least in some cases.
- **Detect**: The function provided by the use case that the arrow originates from that detects the activation of the misuse case that the arrow is directed towards, at least in some cases.

3. Adaptation of Template for Misuse Case

The template suggested by the Rational Unified Process (RUP) contains many of the same entities. Its basic form runs as follows: 1. Use case name, 1.1 Brief description, 1.2 Actors, 2. Flow of events, and so on. Of course, this template is not unique. There are several other templates for use cases, but misuse cases require looking at not only a basic flow, like a use case, but also a second flow. In other words, the point is to see what fields are normally included in use case templates, and then to consider which of these would also be relevant for misuse case templates. Fields in a use case, such as name, basic path, and description, are relevant to both use cases and misuse cases. However, misuse cases assume exceptional events which go against behaviors of use cases. Table 1 shows the template for misuse cases.

Name	Contents
Misuse case Name	Assign a name to the misuse case
Actors	Name of the mal-actor who provokes the misuse case.
Brief description	Summarize a misuse case scenario.
Flow of events	Describe sequentially the basic behavior following this misuse case.
Alternative flow of events	For misuse cases, this occupies a partial event in the basic flow. Alternative flow is also meaningful, although in a lesser way. The alternative path is considered when the basic misuse cases are interrupted by a use case.
Precondition	Describe conditions and backgrounds which are satisfied by triggering the misuse cases and can be ensured by the system itself.
Assumption	Describe conditions which must be true but which cannot be guaranteed by the system itself.
Worst case threat	Describe the outcome if the misuse succeeds. If the misuse case has alternative paths, often this condition will be or contain a disjunction to describe slight variations in outcome.
Capture guarantee	Describe the outcome guaranteed by whatever prevention path is followed. If no prevention path is followed, one might alternatively formulate a wanted prevention guarantee, expressing what one would want the system to achieve with respect to the attempted misuse, but without stating how.
Related business rules	Describe what business rules are broken by each misuse case.
Potential misuse profile	Some kinds of misuse are most likely to be performed by intent whereas other may happen accidentally, for example. Some require insiders or people with enormous technical skill, while others do not.

Stakeholders and threat	This field lists the various stakeholders and their motivations. For misuse cases this slot is even more important. In this field, risks can simply be described textually.
Scope	This field represent the scope of modeling

Table 1. The Template for Misuse Case

4. Non-functionality in Security Requirements

To develop a system, it is first necessary to specify the requirements for the system. So far, the thesis has looked at positive aspects of functions in requirements. But today such one-sided approaches to the functions represented by use cases limit viewpoints on security, reliability and so on. Therefore, the thesis employs a misuse case, which is the negative form of a use case, to document and analyze the security of the system. Alexander defines a misuse case as a use case with hostile intents.[9] In addition to using misuse cases, one can elicit the protective requirements against negative aspects, the so-called security requirements. Security requirements exist because people and the negative agents they create pose real threats to systems. Employing misuse cases and use cases to model and analyze scenarios in systems under design can improve security by helping to mitigate threats. In other words, both use and misuse cases can include subsidiary cases of their own kind, but their relationships to cases of the opposite kind are not simple inclusion. Instead, misuse cases threaten use cases with failure, and appropriate use cases can mitigate known misuse. Table 2 is extracted from Alexander's paper.[10] It shows the rules governing the creation of relationships between use and misuse cases.

[9] I. Alexander, "Misuse Cases: Use Cases with Hostile Intent," *IEEE Software,* January/February 2003, 58-66.

[10] Alexander.

		Source	
	Case Type	Use Case	Misuse Case
Target Case	Use Case	Includes	Threatens
	Misuse Case	Mitigates	Includes

Table 2. Relationship Between Misuse Cases and Use Cases[11]

As illustrated in Table 2, misuse cases can be threats to use cases. For example, the misuse case called "intercept information" can hinder a use case called "send a message" from delivering information. On the other hand, some use cases reduce the threats of misuse cases. For instance, a use case, "check abnormal network," protects from the undesirable behavior of a mal-actor such as a denial-of-service (DoS) attack. Here, one cannot miss that the relations between misuse cases and use cases helps to elicit so-called non-functional requirements. Misuse cases can document the types of nonfunctional or quality requirements that engineers often call the "-ilities": reliability, maintainability, portability and so on.

Exception handling with a use case describes how the system will respond to an undesirable event. Of course, one can employ simple requirement templates to elicit exceptions. But misuse case analysis is also a good way to discover possible exceptions. Misuse case analysis can be a more powerful technique than simply stepping through a template of thinking about exceptions, for several reasons. The analysis involves inverting the problem, taking the negative point of view of use cases, playing games for eliciting misuse cases, visual presentations and so on. In conclusion, products of use/misuse-case analysis that can contribute to effective test planning include specific failure modes, security threats, and exception-handling scenarios.

[11] Alexander.

THIS PAGE INTENTIONALLY LEFT BLANK

II. DECOMPOSITION OF INFORMATION SHARING

A. METHODS OF DECOMPOSITION

Scenarios specified in UML use cases do not capture misuses of systems. Thus, it is necessary to proactively create misuse cases so that requirements for dealing with misuses of a system can be developed. For example, assume that there is a use case named “log on.” Many misuse cases can threaten the use cases in several ways such as “intercept password or ID,” “DoS attack,” “change user profile” and so on. But one cannot recognize when the use case has a problem or where one protects from misuse cases. Therefore, it is necessary to examine use cases in detail. When high-level use cases are decomposed into more detailed cases, many issues must be addressed.

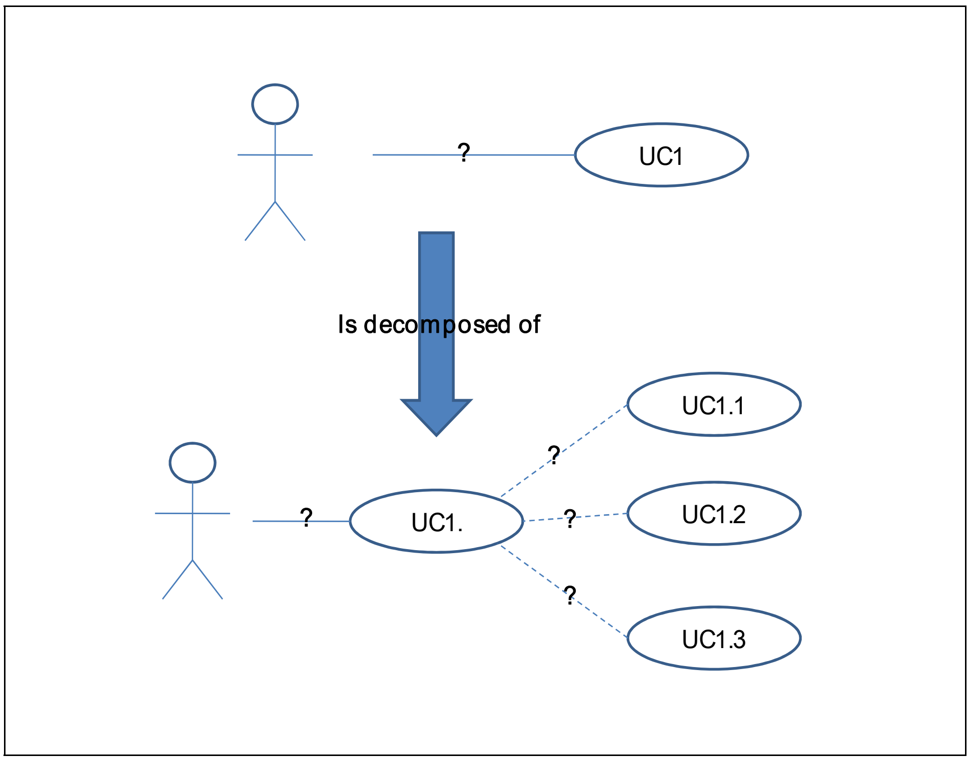

Figure 3. First Iteration of Decomposition

Referring to Figure 3, when high-level use cases are decomposed into detailed sub-use cases, there will be sub-relationships such as “include” and “extend.” This thesis follows the decomposition approach mentioned by Pauli *et al.*[12]

1. Identify candidate cases from textual description.
2. Create initial textual descriptions for each.
3. Identify and model relationship such as “include” and “extend.”
4. Identify and model appropriate actor assignment.

The first step is to identify candidate cases from steps in the textual description of the higher level case. If more details are needed about each step, or if the case does not describe simple behavior, then the step will be chosen to be a decomposed use case. Because future steps rely on the textual step of the decomposed use cases, a textual description is immediately created for these decomposed cases. One then identifies and models “include” and “extend” relationships. The process for identifying and modeling “includes” relationships is necessary so that shared behaviors among the decomposed cases can be accurately and consistently identified and modeled. In addition, the same general process is used for the “extends” relationship to support the textual descriptions by identifying the details of the relationship.

B. DECOMPOSITION OF INFORMATION SHARING

This chapter examines the decomposition of a use case. “Sharing information” is chosen as a main use case. A use case “sharing information” as mentioned above, is concerned with all kinds of information exchange between the requestor and the relevant repository. Sharing information has several features. It could identify whether the requestor is authenticated or not; it could search for a database relevant to the requestor’s query; it could provide information if it finds a match for the query. If there is data related to the request in several repositories, the requestor can receive the data from several repositories. Table 3 shows the flow of events in “sharing information.”

---

12 J. Pauli and D. Xu. “Integrating Functional and Security Requirements with Use Case Decomposition,” 11th IEEE International Conf. on Engineering of Complex Computer Systems. Palo Alto, CA, August, 2006.

Use case Name	Sharing Information
Actors	Requestor
Brief description	A requestor wants to get information from a coalition member. The requestor asks the information broker (IB) to find the information. The IB verifies the identification of the requestor, and then looks for the database relevant to the request in the IB proxy. After finding the database, it returns information from the repository.
Flow of events	1. The requestor requests information to the IB. 2. The IB identifies the requestor. 3. If the IB verifies the requestor, it begins finding databases from its proxy. 4. The IB finds the databases relevant to the request. 5. The IB returns the information from the repositories.
Alternative flow of events	When the IB does not identify the requestor, 1. The IB shows "access denied" message to the requestor and disconnects the network communication.
Precondition	The information broker must have a user's profile. The information broker must have a user's ID and password. The information broker must have the user's public key.
Post-condition	The information broker creates a session for the user.

Table 3. Specification of Sharing Information

1. Identify Requestor

Before a requestor asks for information from the information broker (IB), the requestor must authenticate his or her identity. These procedures precede all other steps. The IB has the IB user's profile which the IB user has already registered. First, the requestor logs on and authenticates to the local IB with his or her ID and password. The IB searches for the user profile and checks to see whether the user profile matches the ID and password. After the IB verifies the user's identity, the IB searches for the user's subject key in the IB subject key repository. If the IB finds the user's subject, the new session is created as a trust credential. When the user's session is created, the IB checks the expiration date for the subject key because the subject is only valid prior to the expiration date of the user's subject key.

Use case Name	Identify Requestor
Actors	Requestor
Brief description	For using the information broker, a user must log on to the IB. The IB has a procedure to give a grant to the requestor to access the IB. The user first puts in his ID and password. The ID and password have to be in the access list in the IB before the user tries to log on to the IB. Then the IB finds the requestor's subject key for verification. If the IB verifies the requestor, the IB creates a one time session for the user.
Flow of events	1. The requestor enters his ID and password when the IB displays the log-on screen. 2. The IB checks the user profile which was stored in the user profile database. 3. The IB finds the matched user ID based on user input. 4. The IB finds the matched user password based on the user input. 5. After checking a basic profile, the IB finds the user's subject key from the IB key storage. 6. The IB creates a session for the requestor and maintains the session until termination of the session.
Alternative flow of events	When the IB does not identify the requestor, 1. The IB shows "access denied" message to the requestor and disconnects the network communication.
Precondition	The information broker must have a user's profile. The information broker must have a user's ID and password. The information broker must have the user's public key.
Post-condition	The information broker creates a session for the user.

Table 4. Specification of a Use Case: Identify Requestor

2. Located Repository Check

After establishing the session, the requestor inputs a query to the IB. Based on the query, the IB searches for the relevant information with the database proxy. After receiving the query from the IB user, the IB analyzes the query and classifies the result of analysis by information taxonomy. Then the IB starts for the information specified in the user's query.

Use case Name	Locate Repository Check
Actors	Requestor
Brief description	The information broker searches for database based on the requestor's query.
Flow of events	1. The information broker user starts to search the database on the basis of the requestor's query. 2. The IB analyzes the query from the requestor. 3. The IB checks the analyzed query to determine whether the query is in the taxonomy of data classification. 4. If the IB finds the matched database, the IB sets up the connection to the repositories.
Alternative flow of events	When the information broker cannot find appropriate classification from the query, 1. The information broker displays the message "specify your query based on information taxonomy." 2. The information broker returns the display to the querying screen. When the local IB cannot find database associated with the query, 1. The local IB forwards the query to the global IB. 2. If the global IB finds the local IB with a database associated with the query, the global IB returns the address to the local IB.
Precondition	1. The information broker must have the user's session. 2. The information broker must have the user's profile. 3. The information broker must have the system of classifying tags.
Post-condition	1. The global or local IB returns the database address to the local IB.

Table 5. Specification of a Use Case: Locate Repository Check

3. Provide Requested Information

After finding databases associated with the query from the requestor, the IB connects to the repositories. The IB checks the permission of the requestor to determine whether the requestor has the correct permission. In order to verify the permission, the IB follows an access list control policy, such as RBAC. If the permission is proved, the IB gathers the fragments of information from several repositories. Then the IB encapsulates the fragments with the IB key, after which the IB supplies the information to the requestor.

Use case Name	Provide Requested Information
Actors	The repository, the requestor
Brief description	After finding the database associated with the requestor, the repository verifies the requester's permission to determine whether the requestor can receive the data. If the permission of the requestor is authenticated, the repository hands over the requested information.
Flow of events	1. The IB finds the database relevant to the requestor's query. 2. The IB verifies that the requestor has permission to receive the data consistent with policy. 3. If it turns out that the requestor has permission to receive the data, the repository approves granting the data to the IB. 4. The IB forwards the data to the requestor.
Alternative flow of events	When the IB cannot verify the requestor's permission to receive the data, the IB displays message "access denied." 1. The IB disconnects the connection to the requestor.
Precondition	1. The information broker must have the user's session. 2. The information broker must have the user's profile. 3. The information broker must have the system of classifying tags.
Post-condition	1. The IB hands over the requested data to the requestor.

Table 6. Specification of a Use Case: Provide Requested Information

C. THE UNIFIED MODELING LANGUAGE BASED ON FUNCTIONS OF INFORMATION SHARING

After decomposition of "information sharing," the process of use case modeling is extended. Figures 4 and 5 show the UML diagrams for decomposition of the use case "sharing information." The requestor logs on and authenticates to the IB. The IB checks the user's identity and creates a session. Next, the user inputs the query to the IB, which in turn searches for relevant repositories. If the IB finds a repository, the IB verifies permission to view data based on the user's role in accordance with policy. Then the IB returns the related data from the repository ciphered with the IB's key.

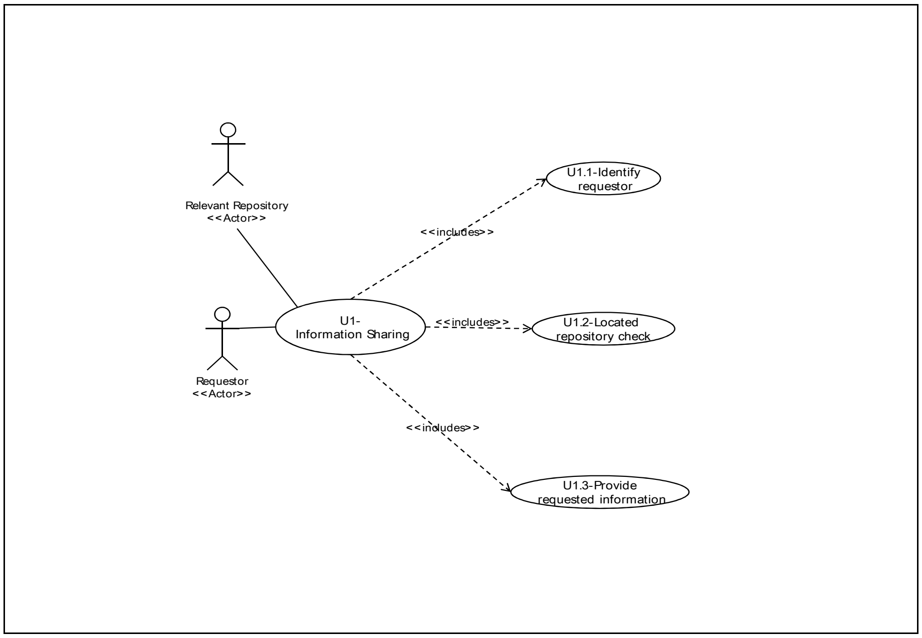

Figure 4. Use Case Diagram: Decomposition of "Information Sharing"

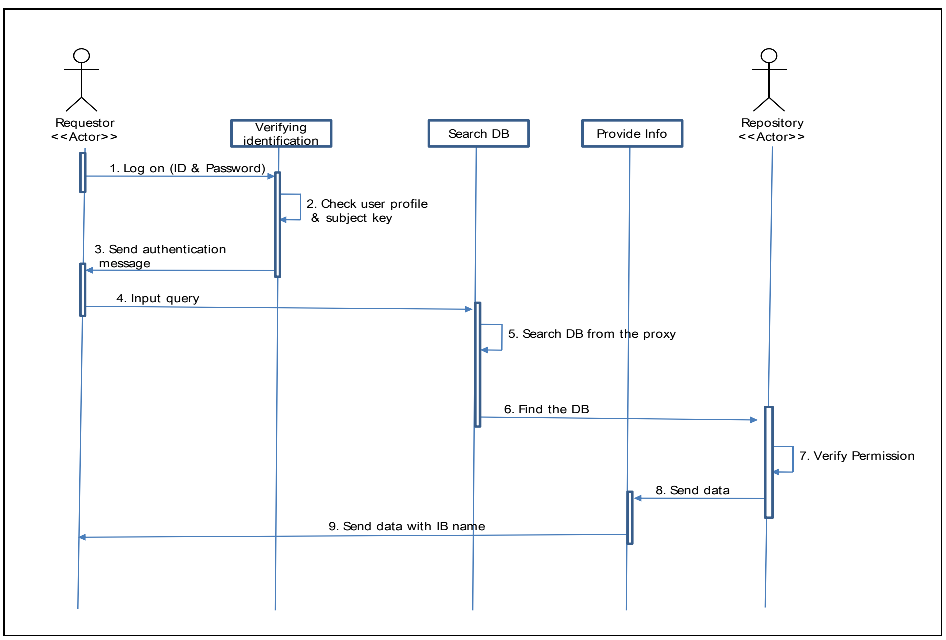

Figure 5. Sequence Diagram: A Use Case of "Information Sharing"

THIS PAGE INTENTIONALLY LEFT BLANK

III. SECURITY PERSPECTIVE ON SHARING INFORMATION

A. ANALYSIS OF MISUSE CASES IN SHARING INFORMATION

The IB provides the requested information to the requestor. Each requestor has both an ID and a password to access to the IB, and the IB stores a requestor's subject key. In addition, the IB is responsible for searching repositories on behalf of the user's role and permission. If the IB finds relevant repositories and verifies the permission from the requestor, it would return the data from the data repository to the requestor. Here, one can find some of the security vulnerabilities, because information should be shared although it may not be secure. "Information sharing" does not have sub-use cases that protect from the foreign access or attack. In other words, misuses from mal-actors can make the IB unavailable, or mistakes of actors can trigger unexpected results.

A misuse case "prevent information sharing" is used as a counterpart to "information sharing." "Prevent information sharing" includes a set of behaviors from a mal-actor such as impersonating the requestor, preventing the IB from finding an information database, preventing the IB from verifying the permission of information, and interrupting flow of information (Table 7 and Figure 6).

Misuse case Name	Prevent Information Sharing
Actors	A hacker
Brief description	A hacker has several ways to interfere with sharing information between the requestor and the opposite side. He could impersonate the user to get information from the IB. He could prevent the IB from finding the location of database based on the requestor's query. He could prevent the IB from verifying a permission of information based on the requestor's role. He could interrupt the flow of information between the requestor and the repository.
Flow of events	1. The hacker accesses the network between user sharing information. 2. The hacker prevents the uses from sharing information a. He impersonates the user to get information from the IB. b. He prevents the IB from finding the location of the database based on the requestor's query. c. He prevents the IB from verifying a permission of information based on the requestor's role. d. He interrupts the flow of information between the requestor and the repository.
Alternative flow of events	When a hacker fails to prevent from sharing information 1. The hacker concentrates on breaking the information broker system. 2. The hacker tries to contact an information broker's system administrator to intercept information.
Precondition	The networks linking the requestor, the IB, and the repositories is connected to the hacker's network physically.
Assumption	Hackers have ability to analyze the signal and code.
Worst case threat	The hacker watches and controls the overall flow of information among the members sharing information.
Capture guarantee	To protect from the hacker's interruption, use more secure and reliable techniques and policy.
Related business rules	The IB cannot guarantee to connect to the IB users.
Potential misuse profile	The hacker is good at hacking the network.
Stakeholders and threat	1 The information broker user 1.1 Loss of trust between the information brokers. 2 The information broker system 2.1 Loss of confidence if security problems get publicized. 3 Database 1.1 Loss of confidence if information is publicized to a unauthorized person.
Scope	The entire information broker environment

Table 7. Specification of a Misuse Case: Prevent Information Sharing

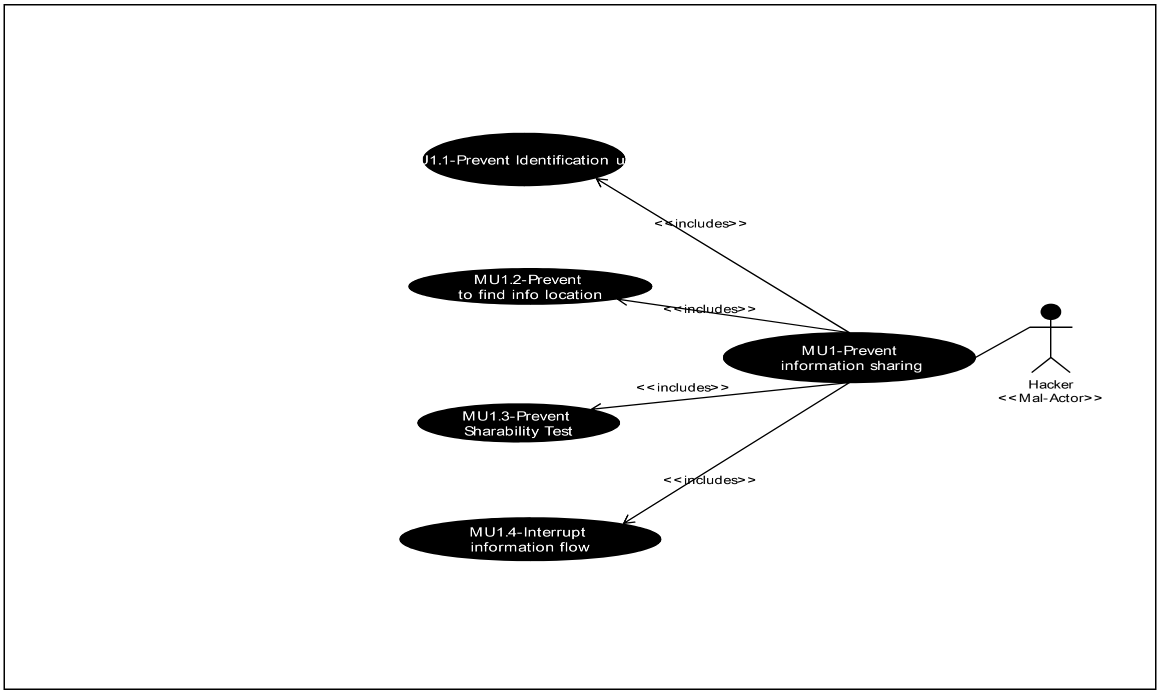

Figure 6. Misuse Case Diagram as "Prevent Information Sharing"

1. Impersonate Requestor

In order to get information, as the first step of using the IB a requestor should log on and authenticate the IB. However, here one assumes that a hacker catches the exchange of identification information between the IB and a requestor, masquerades as the requestor, then receives access to the IB. A hacker could get the requestor's information from the user's computer or by spoofing the data packets between the IB and the requestor when the requestor tries to access the IB. Based on that information, the hacker obtains authentication to access the IB. Table 8 shows how a hacker can impersonate the requestor and enter the IB. Figure 7 presents a sequence diagram of this misuse case.

Misuse case Name	Impersonate Requestor
Actors	Hacker
Brief description	A hacker catches information from a requestor, then tries to access the IB.
Flow of events	1. A hacker gets information from the requestor's PC or by spoofing information between the IB and the requestor. 2. The hacker analyzes the information and tries to access to the IB. 3. The IB grants the hacker authorization to access the IB based on copied information. 4. The hacker acts as if he is a requestor.
Alternative flow of events	When a hacker fails to impersonate a requestor 1. The hacker concentrates on breaking the other information broker system. 2. The hacker attacks the IB or the network by means such as DoS attack or spreading a virus so that the IB user cannot use the IB. 3. The hacker prevents the IB from finding the location of information.
Precondition	The network between the IB and the IB user is connected to the hacker's network physically.
Assumption	Hackers have the ability to analyze the signal and code.
Worst case threat	The hacker pretends to be a requestor and gets information from the IB.
Capture guarantee	To safeguard against a hacker's impersonation, develop checking system for authentication.
Related business rules	The information broker cannot guarantee connection to the IB users.
Potential misuse profile	The hacker has the skill to hack the network.
Stakeholders and threat	1 The information broker user 1.1 Loss of trust between the information brokers. 2 The information broker system 2.1 Loss of confidence if security problems get publicized. 3 Data Repository 1.2 Loss of confidence if information is publicized to a unauthorized person.
Scope	The entire information broker environment

Table 8. Specification of a Misuse Case: Impersonate the Requestor

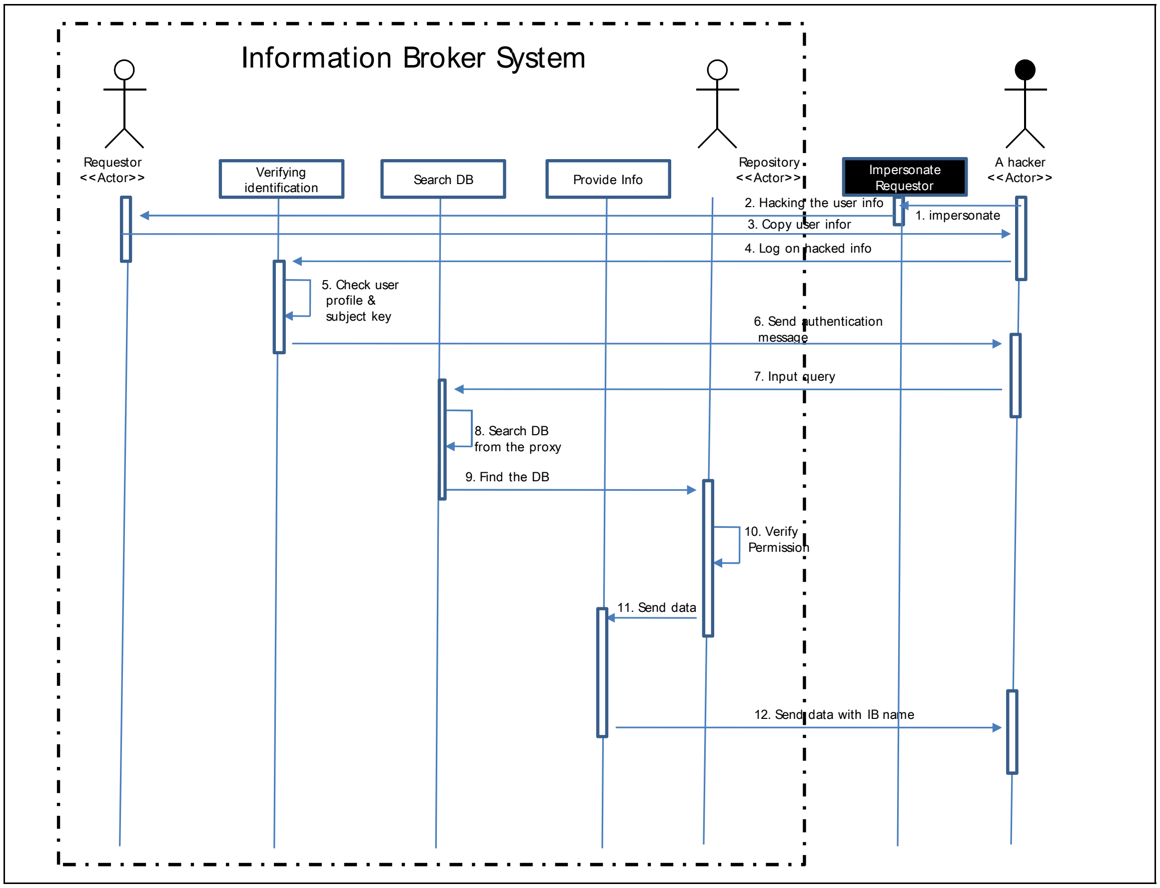

Figure 7. Sequence Diagram of a Misuse Case: Impersonate the Requestor

2. Prevent Finding Info Location

After successfully logging on to the information broker, a requestor inputs a query to get information from the IB. The IB starts to search for databases relevant to the query. Assume that a hacker is trying to prevent the IB from finding the location of a database related to the requestor's query. The first possible attack route is for the hacker to get the IP address of the IB and then break the proxy server storing the addresses of databases. In the second method, a hacker intercepts the query in the network and changes the query into what a hacker wants. Those interruptions prevent the IB from locating a database. Table 9 describes the misuse case "prevent finding info location." Figure 8 depicts the sequencing for the misuse case.

Misuse case Name	Prevent finding info location
Actors	A hacker
Brief description	A hacker does not want the IB to locate a database. Therefore, a hacker tries to break the database proxy in the IB or intercepts a query from the requestor to the IB and changes the query. The requestor cannot receive data which the he wants.
Flow of events	1. The hacker occupies the network between the IB and a requestor. 2. The hacker penetrates the IB and finds a proxy in charge of database. 3. The hacker injects malicious code or virus to break the proxy in order to change the user query. 4. The IB user receives data which is different from what he requested from the IB.
Alternative flow of events	When a hacker fails to prevent the IB from finding databases matched the requestor's query, 1. The hacker concentrates on breaking the other information broker system. 2. The hacker targets the IB or the network with DoS attack or virus to prevent the user from using the IB.
Precondition	1. The network between the IB and the IB user is connected to the hacker's network physically. 2. The hacker occupies the network.
Assumption	Hackers have the ability to analyze the signal and code.
Worst case threat	The hacker breaks the proxy. As a result, nobody can use the IB.
Capture guarantee	To protect from the hacker's interception or penetration, use more secure access control policy.
Related business rules	The requestor cannot trust communication with the IB.
Potential misuse profile	The hacker has the skill to intercept the code and modify the code.
Stakeholders and threat	1 The information broker user 1.1 Loss of trust between the information brokers. 2 The information broker system 2.1 Loss of confidence if security problems get publicized.
Scope	The entire information broker environment.

Table 9. Specification of a Misuse Case: Prevent Finding Database

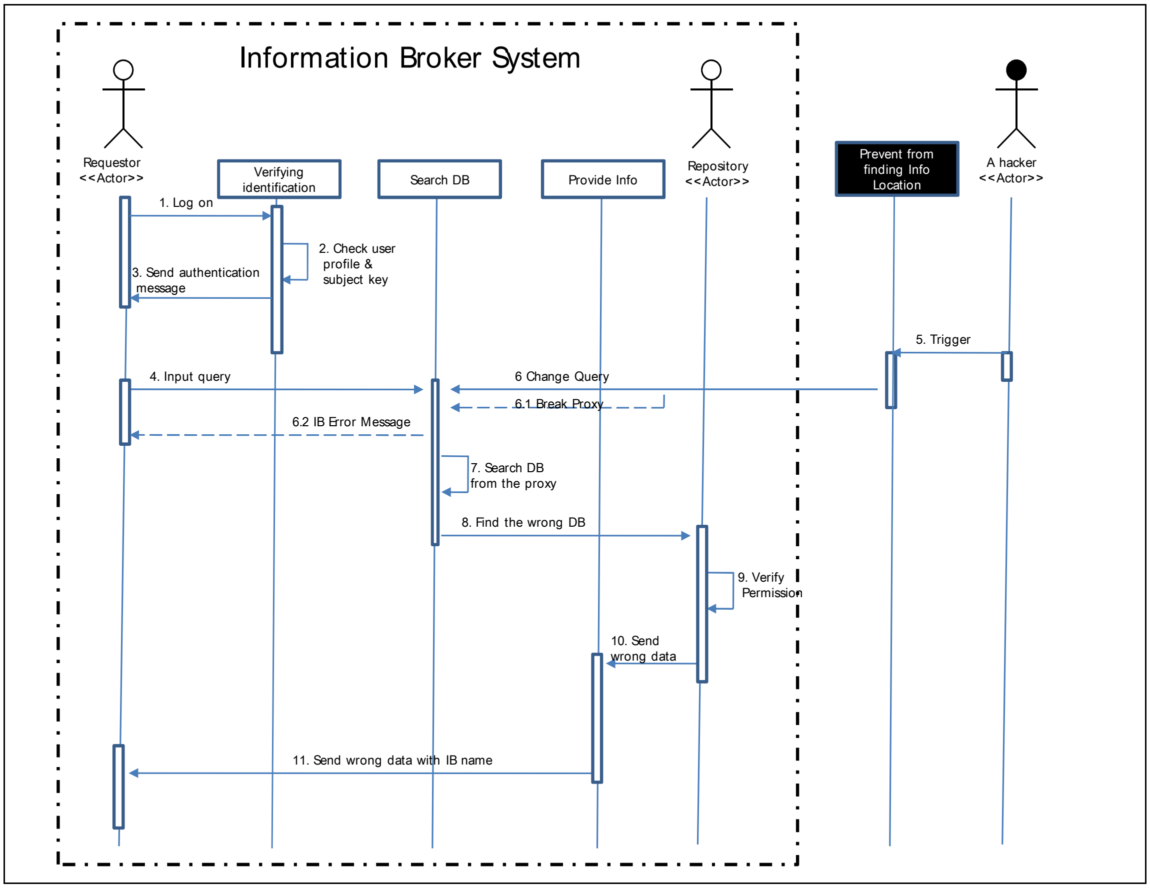

Figure 8. Sequence Diagram of a Misuse Case: Prevent Finding Database

3. Prevent from Verifying Permission

After locating a database, the IB verifies the user's permission based on the user's information and query. However, what if a hacker prevents the IB from verifying the permission to access the information? Exchanging information follows an access control policy. It could be mandatory access control (MAC) or discretionary access control (DAC). Here, the assumption is that this IB uses a DAC policy for flexibility of roles. DAC is based on the user's identity and access control rules. DAC has the following features: (1) Users can protect what they own; (2) Owner may grant access to others; (3) Owner may define the type of access given to others. This reveals the security risks of DAC. What if a hacker modifies the ownership of data? What if a hacker prevents the owner from editing the type of access? Eventually, nobody can get permission to access data. Table 10 and Figure 9 shows a hacker preventing verification of permission to access data.

Misuse case Name	Prevent from Verifying Permission
Actors	A hacker
Brief description	A hacker does not want the IB to verify a permission to receive data from the repository. Therefore, the hacker masquerades an owner of data, then modifies the type of access of data. As a result, neither the requestor nor the original owner can use the data.
Flow of events	1. The hacker occupies the network between the IB and a requestor. 2. The hacker obtains the address of database. 3. The hacker penetrates the database and finds the data matches what the requestor wants. 4. The hacker masquerades an owner of data, then modifies the type of access of data. 5. Neither the requestor nor the original owner from the repository can use the data.
Alternative flow of events	When a hacker fails to prevent the IB from verifying a permission of data from the repository 1. The hacker concentrates on breaking the other information broker systems. 2. The hacker interrupts the IB or the network using a DoS attack or a virus so that the IB user cannot use the IB.
Precondition	1. The network between the IB and the IB user is connected to the hacker's network physically. 2. The hacker occupies the network.
Assumption	Hackers have the ability to analyze the signal and code.
Worst case threat	The hacker breaks the proxy. As a result, nobody can use the IB.
Capture guarantee	To protect from the hacker's interception or penetration, use more secure access control policy.
Related business rules	The requestor cannot trust communication with the IB.
Potential misuse profile	The hacker has a skill to intercept the code and modify the code.
Stakeholders and threat	1 The information broker user 1.1 Loss of trust between the information brokers. 2 The information broker system 2.1 Loss of confidence if security problems get publicized.
Scope	The entire information broker environment

Table 10. Specification of a Misuse Case: Prevent Verifying Permission

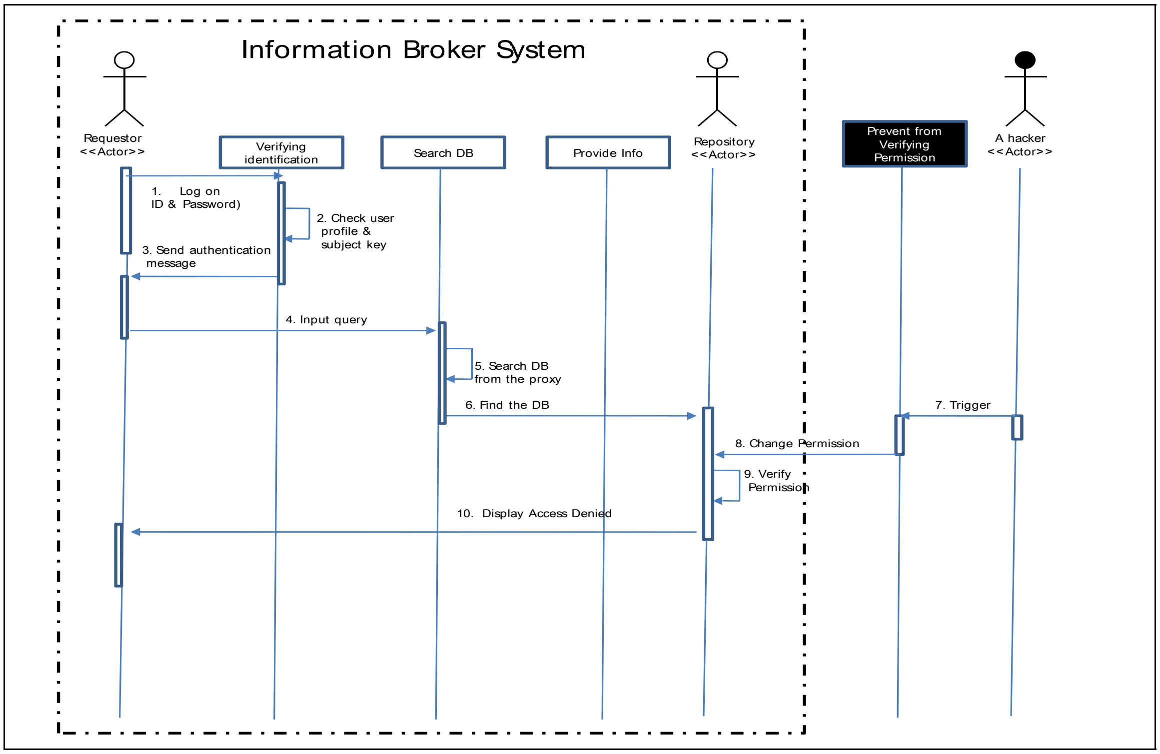

Figure 9. Sequence Diagram of a Misuse Case: Prevent Verifying Permission

4. Interrupt Flow of Information

The main problem of sharing information comes from hacker attacks. Assume that a hacker tries to occupy the network between the IB and the IB user. Once the hacker has the ability to control the network, hacking is much easier. Table 11 and Figure 10 show the hacker occupying the network.

Misuse case Name	Interrupt Flow of Information
Actors	A hacker
Brief description	A hacker observes flow of information on the network and occupies the network.
Flow of events	1. The hacker operates several servers to search for the network between the IB and the requestor. 2. If the hacker finds the network, he analyzes the signal and code. 3. The hacker breaks the code used in communicating between the IB and IB user. 4. After analyzing the signal and code, the hacker tries to control the network.
Alternative flow of events	When a hacker fails to occupy the network, 1. The hacker concentrates on breaking the other information broker systems. 2. The hacker attacks the IB or the network using DoS attack or virus so that the IB user cannot use the IB.
Precondition	The network between the IB and the IB user is connected to the hacker's network physically.
Assumption	Hackers have the ability to analyze the signal and code.
Worst case threat	The hacker totally views and controls the flow of information between the IB and the requestor.
Capture guarantee	To protect from the hacker attack, use a more secure and reliable network like a private network between the IB user and the IB, or encrypt messages.
Related business rules	The information broker cannot guarantee connection to the IB users.
Potential misuse profile	The hacker has the skill to hack the network.
Stakeholders and threat	1 The information broker user 1.1 Loss of trust between the information brokers. 2 The information broker system 2.1 Loss of confidence if security problems get publicized. 3 Database 3.1 Loss of confidence if information is publicized to a unauthorized person.
Scope	The entire information broker environment

Table 11. Specification of a Misuse Case: Interrupt Flow of Information

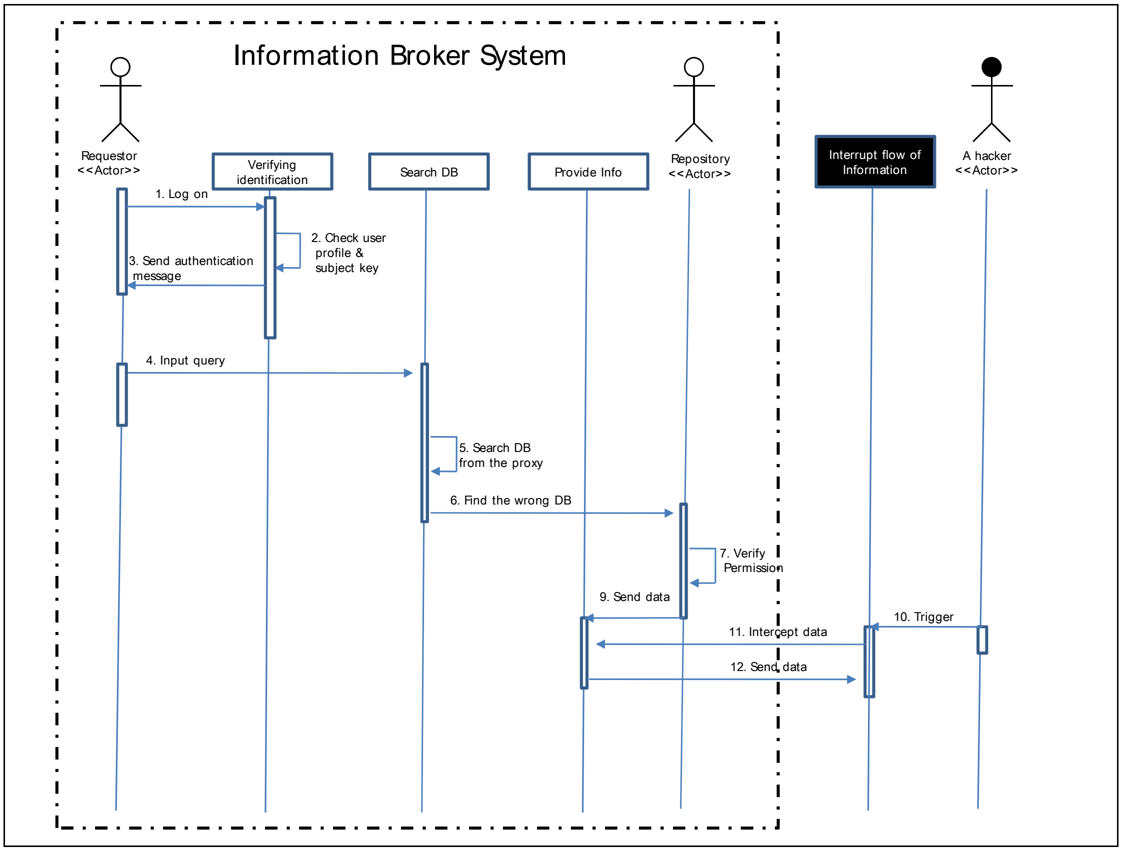

Figure 10. Sequence Diagram of a Misuse Case: Interrupt Flow of Information

The thesis presents research on misuse cases over the operation of the IB, a requestor, and a data repository. Security risks are present in all information sharing procedures. For the flow of information, the proper timing for the exchange of information and keeping the integrity of information are important. However, misuse cases interfere with the timing and integrity of information communication between the IB and requestor. Hackers can intercept the flow of information in the network and modify data in accordance with their own intentions. The next section explores the relationships between these use and misuse cases.

B. RELATIONSHIP BETWEEN MISUSE CASES AND DRBAC USE CASES

Misuse cases of "sharing information" have been described. Those misuse cases are concerned with preventing interference by a hacker. Figure 11 presents an integrated use cases and misuse cases diagram. The black-colored misuse cases threaten use cases of "information sharing."

A hacker's attack harms the IB directly. A hacker who already occupies the network between the IB and the requestor would be willing to control all the packets flowing in the network. He could see and modify the packets consistent with his own intent. In the worst case, the requestor would trust the modified data and make a plan based on the data. In summary, when the requestor wants to get information from the IB, misuse cases directly or indirectly threaten use cases of "information sharing."

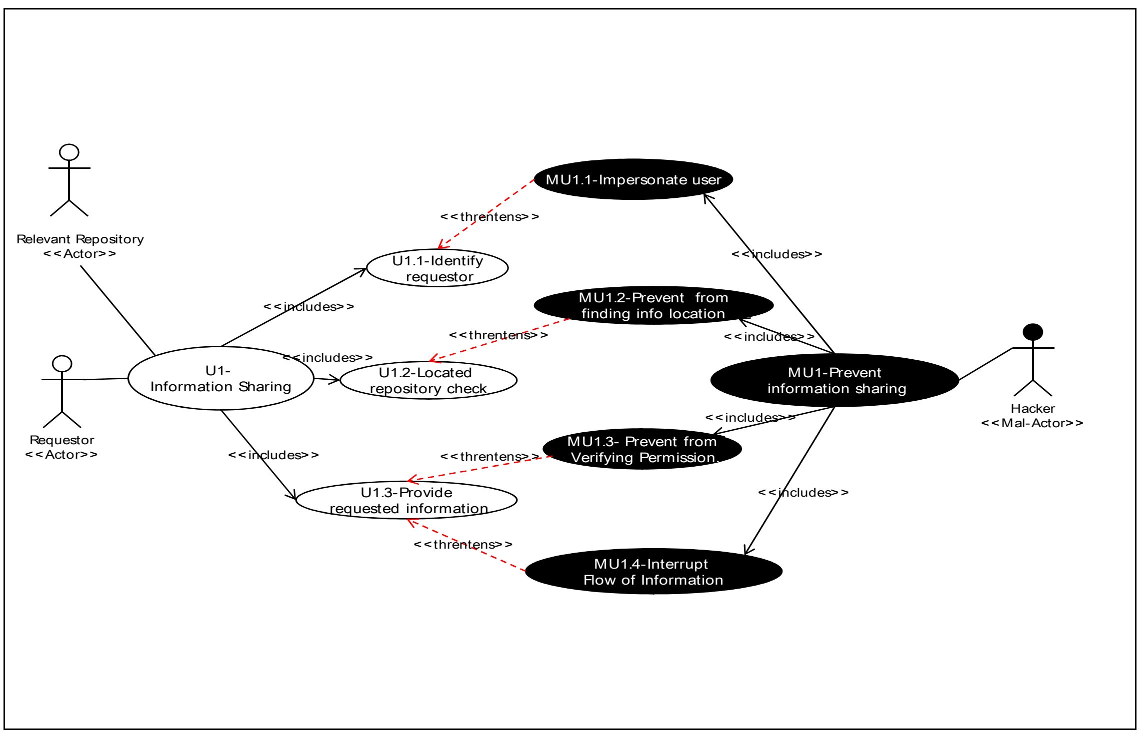

Figure 11. Integrated Use Cases and Misuse Cases Diagram in "Sharing Information"

To make the diagram clearer, identification numbers are assigned to each use case (UC) and misuse case (MU) in Figure 11 and Table 12. Table 12 summarizes which misuse cases threaten the use cases. This result comes from the hacker's activity. Because sharing information depends on the networks among the IB, the requestor, and the repository, it is a loss for a use case "information sharing" if a hacker occupies the network. Therefore, some use cases for security over the misuse cases must be added.

Threaten to =>	UC1	UC1.1	UC1.2	UC1.3
MU1	Threatens			
MU1.1		Threatens		
MU1.2			Threatens	
MU1.3				Threatens
MU1.4				Threatens

Table 12. Integrated Use Cases and Misuse Cases Diagram in Sharing Information

C. SECURITY ENHANCED RECOMPOSITION OF SHARING INFORMATION

The IB is concerned with information sharing. However, there is no way to protect from an external attack. Functions with use cases are decomposed in Chapter II and misuse cases are investigated at the beginning of Chapter III. As the assessment of the relationship between the misuse cases and the use cases shows, the IB might be at risk when a hacker attacks the IB. Therefore, use cases in charge of security over the misuse cases should be added, then the use cases should be recomposed and secured for more secure information sharing.

First, a use case "authentication check" is added against a misuse case "impersonate requestor." Authentication check is in charge of the user identification located in the information broker. Second, a use case "enforce access control policy" is added against a misuse case "prevent from verifying permission." This secure use case is a major part of exchanging information with dRBAC. As mentioned in Chapter I, dRBAC has the advantages of solving the scalability problem and keeping data permission in the IB. Therefore, using dRBAC security policy may be the best way to prevent hackers from conducting a changing permission attack. The third secure use case is “encrypt message.” This use case acts all over the network in the information sharing environment. This thesis focuses on describing this secure use case to prevent a hacker from changing the query and intercepting data packets between repositories and the IB.

1. Authentication Check

In order to protect a hacker from impersonating the requestor, a use case “authentication check” is added to help identify the user. Identification depends on user information such as who you are, what you have, and what you know. Even if a hacker copies the user information such as ID and password from the user's PC, this just satisfies the first and third conditions. But the second condition, what you have, cannot be proved. Therefore, the IB can deny the access from a hacker’s log-on trial. Because the solution is not approached technically, it is described below.

Use case Name	Authentication Check
Actors	The information broker
Brief description	A hacker who has copied the user information tries to log on the IB with the user’s ID and password. The IB sends the embedded module to authenticate the input. The IB finds unique user information and judges an authentication. If it turns out that the approach is from unauthorized person, it shows the message “access denied.”
Flow of events	1. A hacker hacks a user's PC. 2. The hacker tries to log on the IB based on hacked ID and password. 3. The IB sends the ID and password to the check-authentication module. 4. The IB finds matching unique information in the module, and judges whether the person is authorized. 5. If the person does not verify the unique information, the IB regards the access as hacking. 6. The IB denies the access.
Precondition	1. The information broker must have the user’s session. 2. The information broker must have the user’s profile. 3. The information broker must have the system of classifying tags.
Post-condition	1. The IB shows the message “access denied”

Table 13. Specification of a Secure Use Case: Provide Requested Information

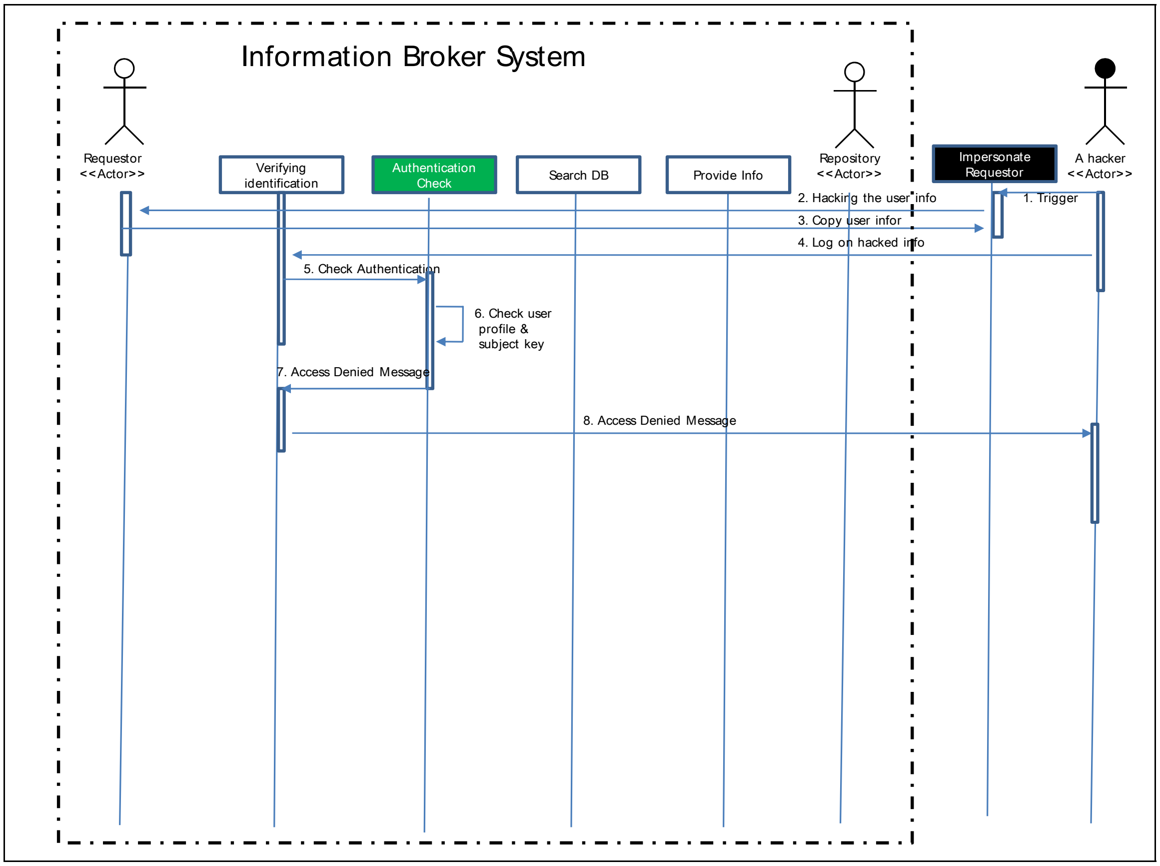

Figure 12. Specification of a Secure Use Case: Authentication Check

2. Enhance Secure Access Control Policy

The dRBAC is chosen as a secure access control policy in secure use case "enhancing secure access control." Security is enhanced with dRBAC, which delegates and proves authentications like RBAC but is superior because it is scalable. It is important to know how a repository grants permission to the requestor. This results from the delegation of rules.

The dRBAC model shows three different types of delegation models: self-certifying, third party, and assignment. The first two permit an entity to delegate permissions associated with a role, either in its own namespace or in another, while the third permits delegation of the "right of assignment" of the referenced role. Basically, role delegation is defined as signed certificates that extend access rights on some object to a subject: “[subject → object] issuer”, where object is a role, issuer is an entity, and subject is a role or an entity. Here, “→” means “has a permission of.”

a. Self-certified Delegation: An issuer A grants role A.a to some subject. The role granted is defined within A’s name: “[subject → A.a]A”

b. Assignment Delegation: Entity B grants some subject the right to delegate role A.a to others. The tick indicates that the Subject can further delegate the role. B and A may or may not be the same entity: “[subject → A.a’]B

c. Third-Party Delegation: In third-party delegation, some issuer B exercises its right to delegate a role defined in A’s namespace. A and B are not the same entity: “[subject → A.a]B”

For example, assume that Kim is an agent in the NIS and Bob is an agent in the CIA. The NIS agent Kim will take advantage of a coalition between NIS and CIA to obtain data access through the CIA. Table 14 shows the delegation authorizing this access.

Sequence	Delegation
1	[Kim → NIS.agent] NIS
2	[NIS.agent → CIA.agent with CIA.picture = area 5027] Bob
3	[Bob → CIA.agent] CIA
4	[CIA.agent → CIA.database with CIA.picture =’] CIA
5	[CIA.database → CIA.access with CIA.picture = area 5027] CIA

Table 14. Delegations Supporting Kim’s Access to CIA Resource

Delegation (1) identifies Kim as NIS.agent. Delegation (2) defines the coalition between NIS and CIA as set up by the IB, whose authorization for doing so is provided by delegations (3)-(5).

Use case Name	Enhance Secure Access Control Policy
Actors	The information broker
Brief description	After finding the database associated with the requestor, the repository verifies a permission of the requester whether the requestor can receive the data or not. If the permission of the requestor is authenticated, the repository would hand over the requested information by dRBAC.
Flow of events	1. The IB found the database relevant to the requestor's query. 2. The IB verifies that the requestor has the permission of the data by dRBAC policy. 3. If it turns out that the requestor has a permission of the data, the repository approves granting the data to the IB. 4. The IB forwards the data to the requestor.
Alternative flow of events	When the IB cannot verify the permission of the data the requestor has, 1. The IB displays message to the requestor, "access denied." 2. The IB disconnects the connection to the requestor.
Precondition	1. The information broker must have the user's session. 2. The information broker must have the user's profile. 3. The information broker must have the system of classifying tags.
Post-condition	1. The IB hands over the requested data to the requestor.

Table 15. Specification of a Secure Use Case: Enforce Security Access Control Policy

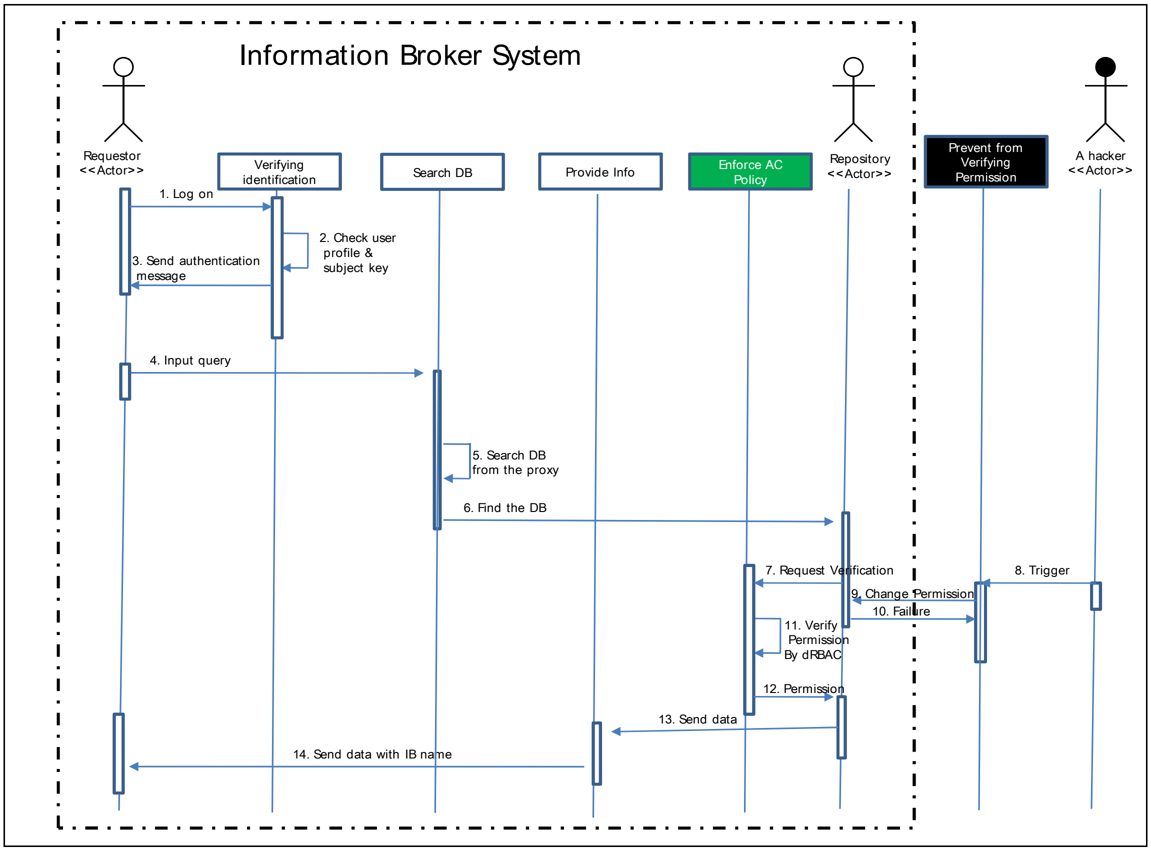

Figure 13. Sequence Diagram: Enforce Security Access Control Policy

3. Encrypt Messages

When a requestor sends the query to the IB, or the IB and the repository sends data related to the user's query, it is important to maintain both the integrity and confidentiality of information flow. A hacker can intercept the packets passing between IB members and modify the data, increasing security risks. To protect from those kinds of attacks, a secure use case "encrypt message" is added. Table 16, Figure 13 and Figure 14 describe the use case "encrypt message."

Use case Name	Encrypt Message
Actors	The information broker
Brief description	When the user inputs the query or data packets are moving among the IB members, the security risk is increased. Encrypting messages guarantees a secure network.
Flow of events	1. The requestor inputs the encrypted query to the IB. 2. The IB receives the message and analyzes the query and finds the database related to the encrypted message. 3. After verification, the repository sends encrypted data. 4. The IB re-encrypts the message with IB name.
Precondition	1. The information broker must have the user's session. 2. The information broker must have the user's profile. 3. The information broker must have the system of classifying tags.
Post-condition	1. The IB hands over the requested data to the requestor.

Table 16. Specification of a Secure Use Case: Encrypt message

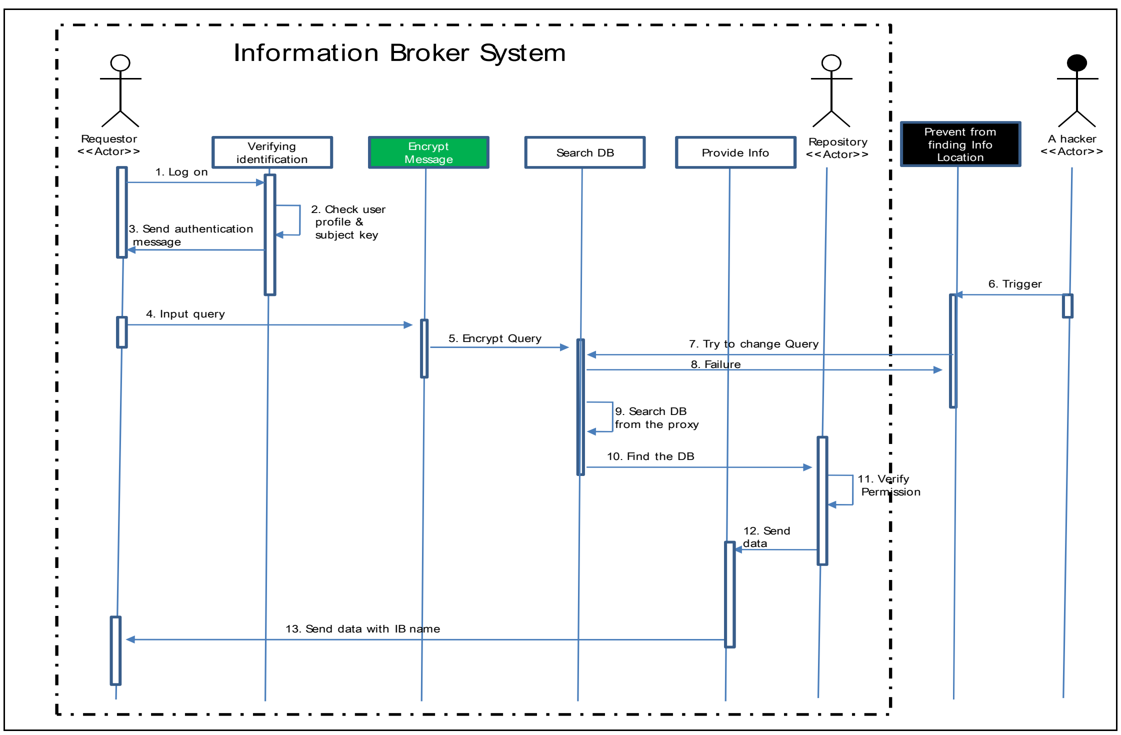

Figure 14. Sequence Diagram of "Encrypt Message": Encrypt the Query

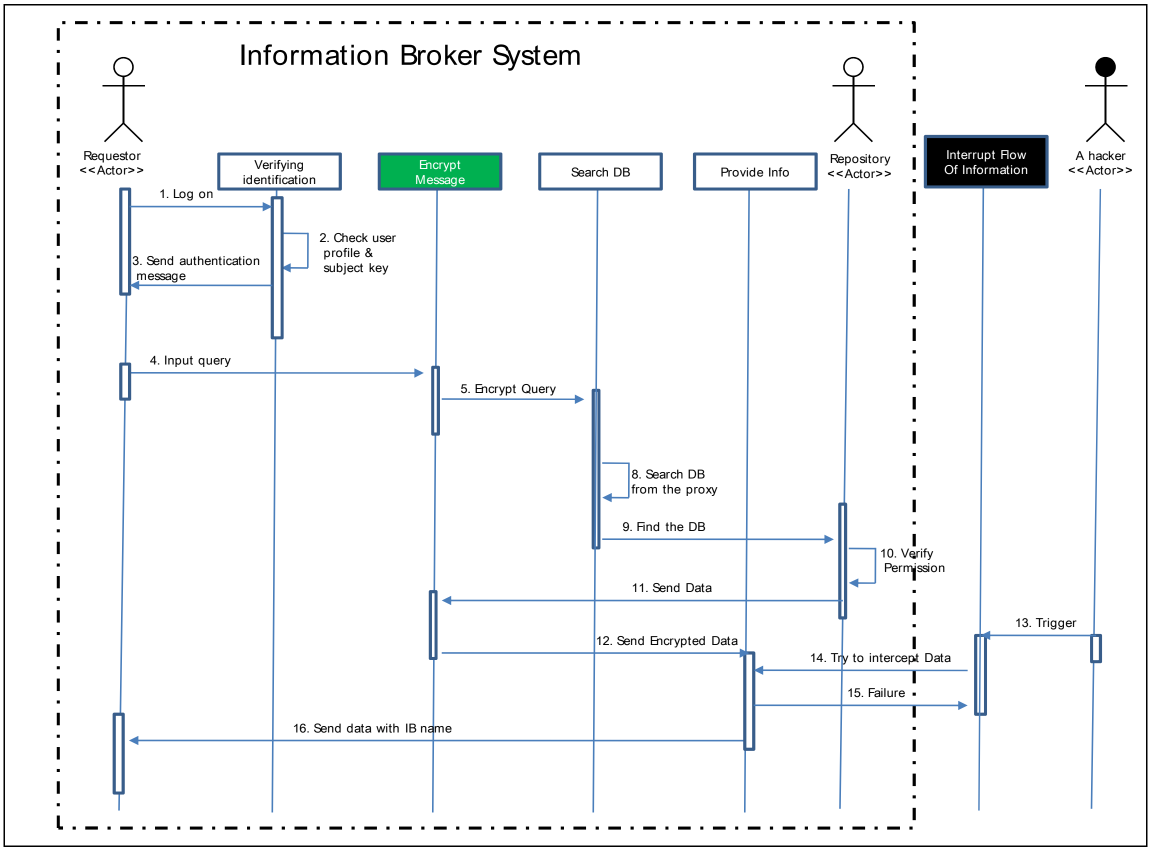

Figure 15. Sequence Diagram of "Encrypt Message": Encrypt the Data

D. SUMMARY

Through decomposition and re-composition of use cases and misuse cases, a new model has been developed. Figures 16 and 17 present a sequence diagram and a use case diagram of misuse cases and secure-enhanced "sharing information" use cases. Table 17 indicates the relationship between misuse cases and secure-enhanced use cases. SU1 (Secure Use Case 1) "authentication check" detects the misuse case "impersonate requestor." SU2 "enhancing access control policy" is concerned with dRBAC policy, which is used for exchanging information between users. SU3 "encrypt messages" also helps to protect against attempts to intercept and change data packets.

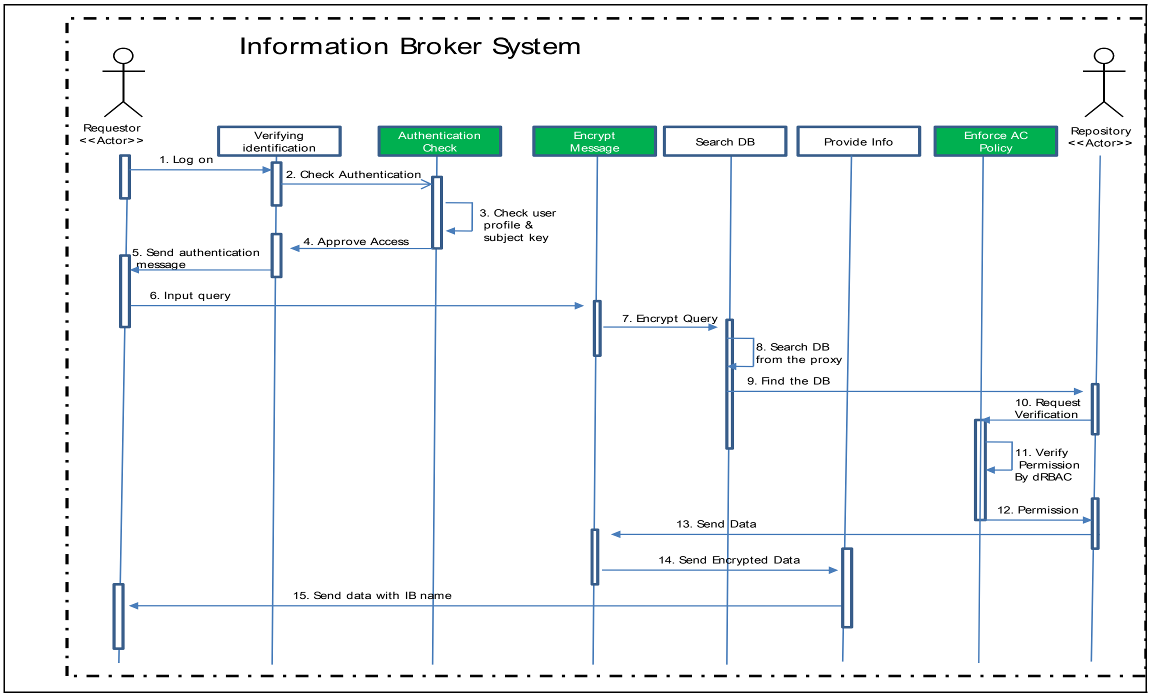

Figure 16. Sequence Diagram of Misuse Cases and Secure-enhanced Use Cases

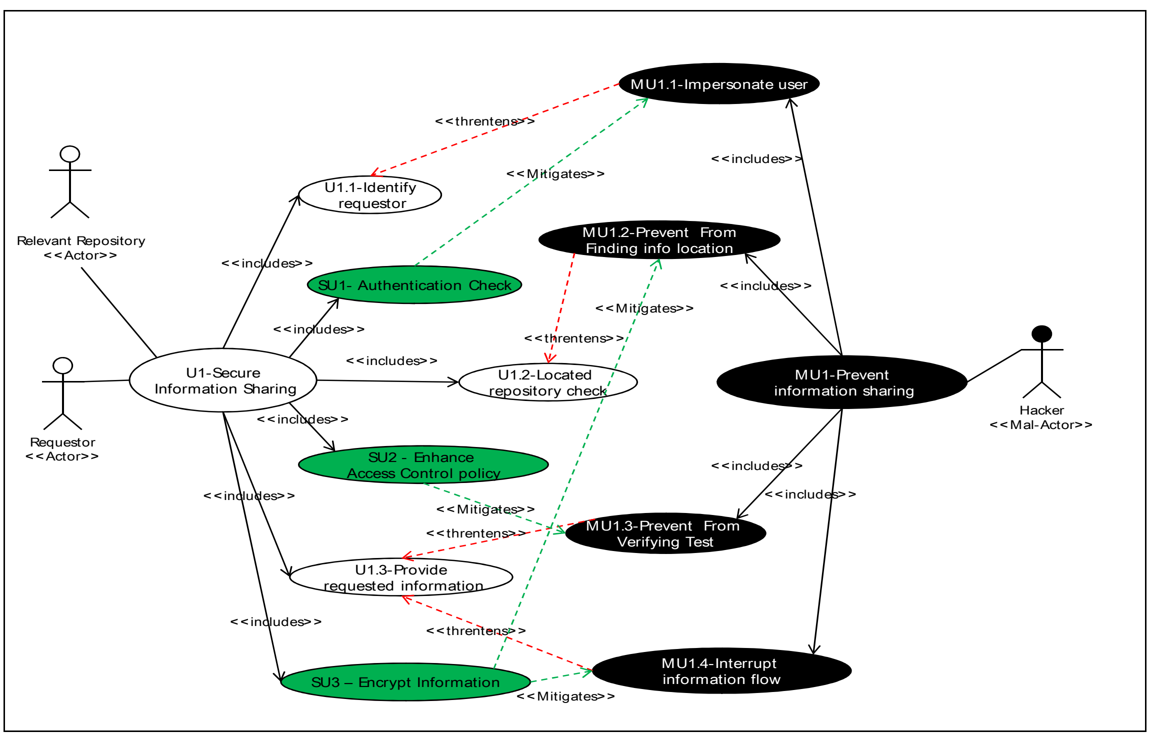

Figure 17. Use Case Diagram of Misuse Cases and Secure-enhanced Use Cases

	UC1	UC1.1	UC1.2	UC1.3	SU1	SU2	SU3
MU1	Threatens						
MU1.1		Threatens			Mitigates		
MU1.2			Threatens			Mitigates	
MU1.3				Threatens			Mitigates
MU1.4				Threatens			Mitigates

Table 17. The Relationship between Misuse Cases and Secure-enhanced dRBAC

With added use cases for security, security risks, which are discovered by misuse cases, can be reduced. This approach is best used during the requirements specification and analysis phase of development for the IB system where use cases can be used to document the requirements. In addition, early consideration of security helps to assure the user will not make a mistake operating the IB. Moreover, if the IB has a problem with a function, the function can be traced and the problem found easily because all the dRBAC functions have been covered in detail.

IV. IMPLEMENTATION OF THE CASE STUDY

A. MODEL OF SCENARIO IN DRBAC

This section presents the flow of data and roles in a case study about the detection of nuclear weapons or technology enabling the production of nuclear weapons in North Korea. The flow of data and roles are described in Figure 18.

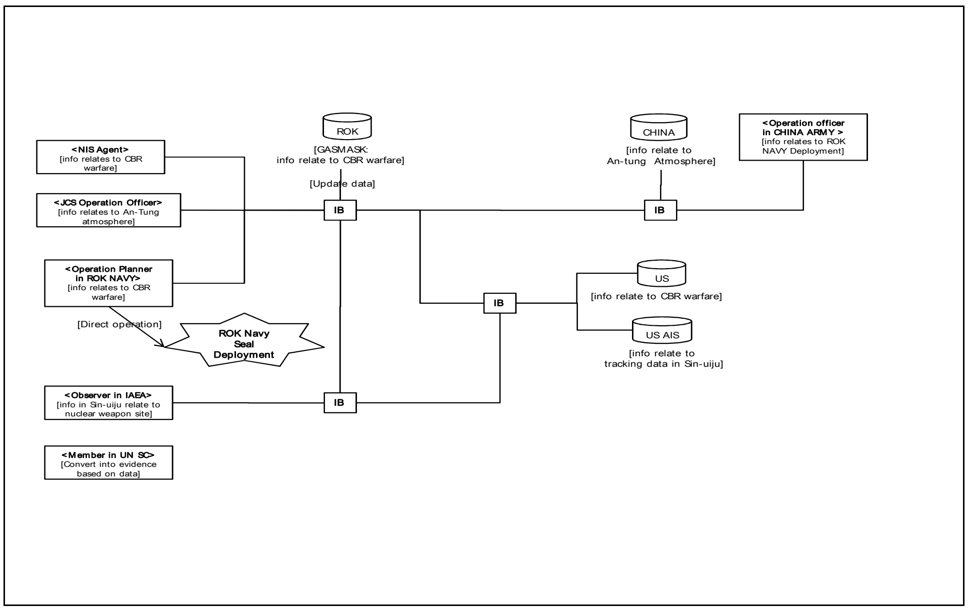

Figure 18. Flow of data and roles

1. Background

The case study is built on the weapon detection problem involving North Korea. North Korea poses a threat to North America, Europe, and Asia in this scenario, which assumes that North Korea's nuclear weapons program is detected by a military intelligence agency.

2. Phase 1 – Reconnaissance and Detection

a. Situation

The ROK reconnaissance satellite, Uribyul, in orbit over North Korea, detects suspicious objects (e.g., launchers and other missile-related facilities) in the vicinity of the border area between North Korea and China known as Sin-uiju. Imagery from the satellite is sent to the Korean NIS. The pictures are studied by intelligence analysts. The analysts suspect that the site hosts nuclear weapons.

- Actor 1. Baek, a NIS agent in Seoul, has a user profile that authorizes access to data labeled SECRET and marked Law Enforcement Sensitive.
- Actor 2. Kim, a ROK Joint Chief of Staff (JCS) operations officer, has a user profile that authorizes assess to data labeled TOP SECRET.

b. Scenario

(1) Step 1. The NIS agent (actor 1) logs on and authenticates to the IB. The NIS agent accesses a shared workspace, code-named Gasmask, established for data related to chemical, biological and radiological (CBR) warfare. This workspace includes data labeled SECRET, SECRET-NOFORN, Sensitive but Unclassified, and Law Enforcement Sensitive level.

(2) Step 2. The NIS agent rechecks satellite images in the workspace and finds an image of the territorial feature where nuclear weapons can be launched, in addition to images of launchers and other suspicious objects.

(3) Step 3. The NIS agent remembers seeing a similar territory on the CBR database of suspected launch sites and makes a query to the information broker requesting information relating to CBR warfare. The information broker returns images and analysis reports showing suspected CBR facilities at that site. (Because the database in South Korea does not have a lot of data on hand relating to CBR warfare, the NIS

requests such data from the U.S.) The NIS agent downloads the files, which are labeled SECRET-Releasable China, and then posts them to the Gasmask shared workspace.

(4) Step 4. The ROK JCS operations officer (Actor 2) logs on and authenticates to the information broker. The operations officer makes a query to the information broker requesting the recently analyzed information about the air quality of Sin-uiju from the State Environment Protection Administration (SEPA- Chinese Environment Bureau) database in An-Tung, which lies on the Chinese side of the border near Sin-uiju. The information broker returns the data about the atmosphere around An-Tung that includes information on Sin-uiju. The operations officer downloads the data and analyzes it with regard to CBR warfare.

(5) Step 5. Based on the SEPA report on heightened levels of radiation in Sin-uiju, the ROK JCS operations officer suspects that nuclear weapons are hidden there. The officer creates the annotated graphic, classified SECRET-Releasable UN, and posts the analyzed data in the shared workspace.

3. Phase 2 – Rechecking and Defense

a. Situation

The members of the UN Security Council recognize the possibility that nuclear weapons exist in North Korea. The UN Security Council directs IAEA observers to verify the graphical data on the Sin-uiju region from the shared workspace Gasmask. They discover abnormal air quality statistics indicative of the presence of nuclear material. The members of the UN Security Council prepare two contingency plans: (a) politically pressure North Korea to scrap the nuclear weapon missiles by itself; or (b) destroy the suspected nuclear missile launching site. For plan (b), the members request that the Minister of Defense in South Korea direct the ROK Navy Command to complete the mission.

- Actor 3. James, a member of the UN Security Council, is directed to prepare for negotiations with North Korea. Actor 3 has a user profile that authorizes access to data and information labeled TOP SECRET.

- Actor 4. John, an IAEA observer, is directed to check information on the possibility of North Korea possessing a nuclear weapon. Actor 4 has a user profile that authorizes access to data and information at TOP SECRET.
- Actor 5. Park, a ROK Navy Command operations planner, is tasked with determining options for destroying the suspected nuclear weapon missile launching site near Sin-uiju. Actor 5 has a user profile that authorizes access to data labeled TOP SECRET.

b. Scenario

(1) Step 1. The IAEA observer (actor 4) logs on and authenticates to the information broker. The observer makes a query to the information broker requesting recent data about the area near Sin-uiju. The information broker returns the following data from the Gasmask shared workspace: (a) the flame-grab image of the suspect region in North Korea; (b) the CBR warfare air quality statistics data file.

(2) Step 2. The IAEA observer logs on and authenticates to the IB. The observer makes a query to the IB requesting the data relating to buildings under construction and tracking data for all of the vehicles near Sin-uiju in the most recent one-year period. The IB returns the tracking data of the automatic identification system (AIS) and the locations of buildings under construction in the vicinity of Sin-uiju during previous months from data in U.S. Naval Intelligence databases.

(3) Step 3. The IAEA observer analyzes and verifies the data. The observer informs the UN Security Council of the possibility that North Korea has nuclear weapons.

(4) Step 4. The UN Security Council member (actor 3) makes a query to the information broker requesting evidence for use in negotiations with North Korea. The information broker finds the requested data in Gasmask and returns the data so that actor 3 can see it. Using the data, the member gathers information, converts it into evidence and formulates a strategy for negotiation.

(5) Step 5. The operations planner in the ROK Navy Command (actor 5) accesses the shared workspace Gasmask and examines the graphical plot of the Sin-uiju region. The planner simulates the possibility of mission failure. If the nuclear weapon explodes during the mission, the emission of radiation would directly affect An-Tung. Therefore, the operations planner recommends that the People's Republic of China be informed of the hazard. Actor 5 also creates a mission for destroying the suspected site in Sin-uiju.

(6) Step 6. The operations planner on ROK NAVY Command overlays the locations of the ROK Navy SEAL force operatinging the Yellow Sea. The operations planner creates a data set showing where the ROK Navy SEAL Force will deploy, marked SECRET-Releasable China.

4. Phase 3 – Supporting Allies

a. Situation

The Minister of the ROK Defense informs the China Department of Defense of the potential threat. The Chinese defense ministry directs China Army Intelligence to (a) evaluate negative effects from failure of the mission; and (b) help the ROK Navy combatants escape safely.

- Actor 6. Wang, an operations officer in China Army Intelligence, has a user profile that authorizes access to data and information found in a number of Chinese databases as well as ROK databases labeled SECRET-Releasable China.

b. Scenario

(1) Step 1. The Chinese operations officer (actor 5) logs on and authenticates to the information broker. The operations officer makes a query to the information broker requesting the data of the locations in the vicinity of the Sin-uiju where ROK Navy Special Forces combatants will deploy. The information broker returns the requested data from the ROK Navy databases. The operations officer downloads the

requested data and displays it on the Chinese Army Intelligence computer. The operations officer makes a plan to help the ROK Navy combatants escape safely after their mission.

(2) Step 2. The China operations officer makes a query to the information broker requesting information on possible effects of radiation in the event that the mission fails. The information broker returns data files relating to CBR effects located in Gasmask. After evaluating the threats of radiation contamination if the mission fails, the operations officer makes a plan to protect people in An-Tung from radioactive emissions.

B. ASSESSMENT FOR THE SCENARIO

The purpose of this section is to demonstrate the technical feasibility of conducting a systematic formal approach using dRBAC. In addition, this section explains why dRBAC is appropriate for modeling information sharing for coalition environments. Security-enhanced use cases for information sharing have been examined in this thesis. Table 18 contains a summary of the roles introduced in the scenario.

Role #	Role	Organization	Description
1	Agent	NIS	Responsible for gathering data from a satellite and analyzing intelligence associated with CBR warfare
2	Operations officer	ROK JCS	Responsible for analyzing intelligence associated with CBR warfare; reports to UN
3	Member	UN Security Council	Preparing for negotiations with North Korea over their nuclear weapons program
4	Observer	IAEA	Checks information on the possibility that North Korea has a nuclear weapon
5	Operations planner	ROK NAVY Commands	Tasked with determining options for destroying the suspected launching site for a nuclear-armed missile near Sin-uiju
6	Operations officer	China Army Intelligence	Responsible for evaluating negative effects if the mission fails and with helping ROK NAVY combatants escape safely.

Table 18. Roles in the Scenario

These scenarios are applied by dRBAC. The proof uses the model proposed by Freudenthal et al., which serves as the basis for analyzing the scenarios. Appendix B contains a description of the syntax of dRBAC policy.

In the first step of sharing information in scenario 1, the NIS agent, Baek, is willing to get CBR information through the IB. The information comes from a CIA database. Table 19 shows the proof of delegations using RBAC policy.

Phase 1, Step 3
(1)[Baek → NIS.agent]NIS
(2)[NIS.agent → CIA.agent
with CIA.images = Korea area
and CIA.document = Korea area
and CIA.secret-level <= Secret] CIA
(3)[CIA.agent → CIA.DBmember
with CIA.images = ‘
and CIA.document = ‘
and CIA.secret-level <= ‘] CIA
(4)[CIA.DBmember → CIA.access
with CIA.images = Korea area
and CIA.document = Korea area
and CIA.secret-level = Secret] CIA
Result
[Baek → CIA.access
with CIA.images = Korea area
and CIA.document = Korea area
and CIA.secret-level = Secret] CIA

Table 19. Delegation of Baek’s Access

Delegation step (1) identifies Baek as an agent in NIS. In step 1, one can check authentication of Baek’s subject key. The IB compares the encrypted key with the subject key. Under the assumption that the key is identified, delegation step (2) defines the coalition between the CIA and NIS as established by the CIA. Step (2) also provides limitations or restrictions as specified by the CIA. These limitations can be set, removed, or modified on a case-by-case basis depending on the strength or desired strength of the coalition. In this case, NIS Baek is authorized to view all the data that a CIA agent is entitled to see and the CIA agent is authorized to provide the delegation of permissions as described in delegation steps (3) - (4).

Figure 18 is a distributed proof construction of all of the steps in the dRBAC process. This case study starts off with agent Baek from NIS establishing a connection to a CIA server to access information (Step 1). In this case the coalition role of "NIS" authenticates itself to CIA using a public-key cryptographic protocol and requests access to the data on Baek's behalf by passing on delegation (1) which validates Baek as a NIS agent. To authorize access, the CIA server must find a proof for NIS.agent → CIA.access. Here, "→" means "have a permission of." When combined with delegation (1) it provides that Baek is authorized access to applicable CIA data (Baek → CIA.access).

The CIA server queries its trusted local wallet for the requested proofs as seen in step 2. If it fails to find the proofs locally, the wallet attempts to discover the delegations necessary to build the proof. The wallet will contact the home wallet corresponding to the role NIS.agent, issue a query, and discover that there is a defined relationship between the roles NIS.agent and CIA.agent. The server wallet now has a chain from Baek to CIA.agent. There is still an outstanding requirement that would authorize CIA agents to the CIA database (CIA.agent → CIA.access). A direct query is issued for a subject to object search involving CIA.agent → CIA.access (Step 4).

The results of the query are a self-certified delegation. This provides proofs showing that Baek has access to CIA data (Baek → CIA.access). In step 5, "[d]elegations from this proof are inserted into the local wallet, which is trusted to verify signatures and establish its own validation subscriptions."[13] At this point, limitations and restrictions can

[13] Freudenthal *et al*., 412.

be placed on access to data. In step 6, the proof is returned to the original requester and stored as an object. This object allows for the continuous monitoring of delegations authorizing Baek's access. Such continuous monitoring could implement the temporal aspects of RBAC, providing limited access to data based on time.

The same methods are used to prove the rest of permissions in information sharing using dRBAC, as shown in Appendix B.

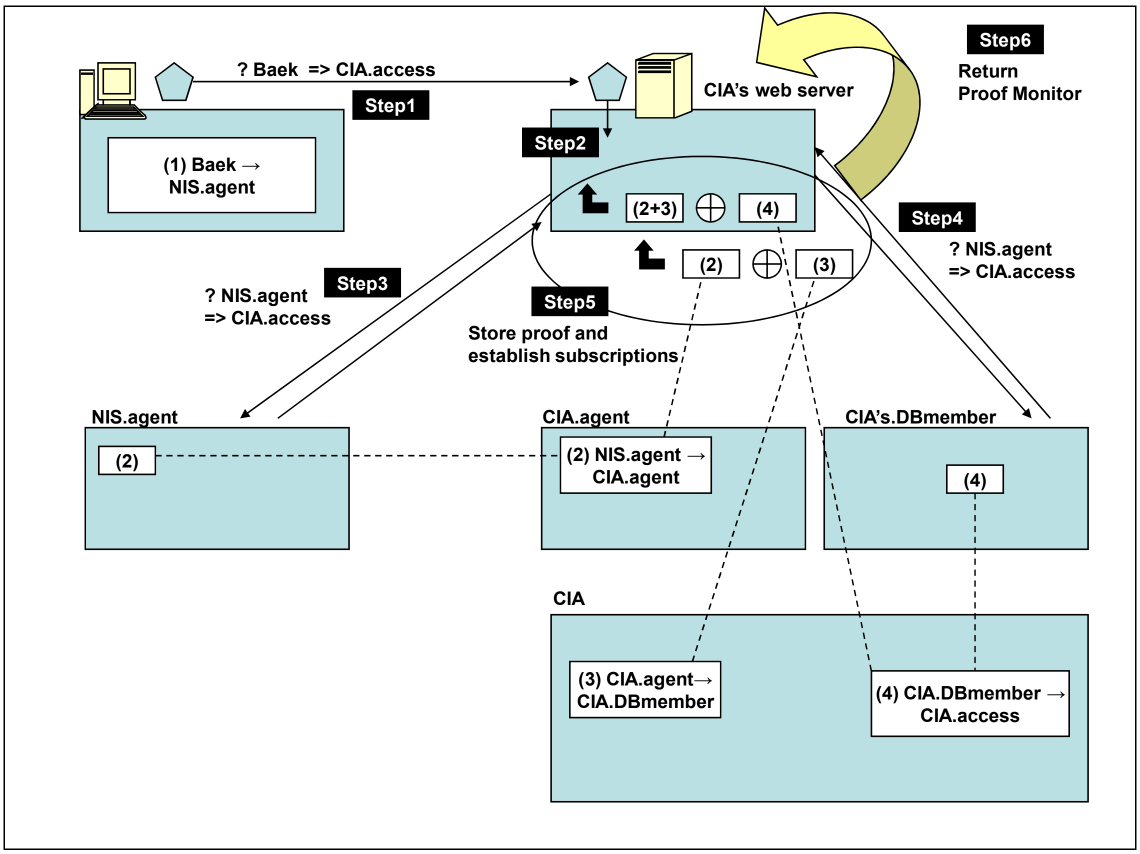

Figure 19. Distributed Proof Construction of Baek's Access in Step 3, Phase 1

THIS PAGE INTENTIONALLY LEFT BLANK

V. CONCLUSION AND RECOMMENDATIONS

This thesis demonstrates an approach applying misuse cases to drive the specification of requirements for secure information sharing.

A. CONTRIBUTIONS OF THIS THESIS

The thesis presents an integrated system development and risk management process for development of secure information sharing. The proposed process consists of decomposing the main functions and analyzing their risks using misuse cases. Chapter III shows that the detection and mitigation of security obstacles do not require complex technical skills because use cases are focused on a high level of abstraction. In addition, the iterative development with sequence diagrams and secure-enhanced use case diagrams facilitates understanding of how the mitigation should be implemented.

The research indicates that it is best to take an incremental approach to applying the dRBAC principle to solve a “need to share” problem. The dRBAC supports dissemination within a coalition of information. The roles created for coalition partners can have set limitations to control access to the information that they need to share. The creation of roles specific to a particular event or operation will be the baseline for controlling the flow of information to coalition partners.

B. RECOMMENDATIONS FOR FUTURE WORK

One area of future work is automated tool support. For example, a traceability tool for managing the relationships among use cases, misuse cases, and other system artifacts. Another area of further research to consider is that of metrics, specifically, metrics that capture the action of the information returned to the requestor. In addition, what types of feedback mechanisms can be put in place to assist in improving the action and overall quality of service provided by the information broker.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. SYNTAX FOR THE BASE DRBAC DELEGATION MODEL

Entities: A public key that represents a principal or a resource and defines a namespace that can contain roles.

Form: cryptographic public key and a human-readable name.

Example: Kim; NIS

Role: A name within an Entity's namespace.

Form: Entity.LocalName

Example: NIS.agent

Role Delegations: Signed Certificates that extend access rights on some object to a subject. Access to an object by a subject can be extended by the issuer of the certificate.

Form:[Subject → Object] Issuer

dRBAC includes three major types of delegations:

Self-certified Delegation: An Issuer A grants role A.a to some Subject. The role granted is defined within A's namespace.

Form:[Subject → A.a]A

Example: [Kim → NIS.agent]NIS

Assignment Delegation: Entity B grants some Subject the right to delegate Role A.a to others. The tick (') indicates that the Subject can further delegate the Role. B and A may or may not be the same entity.

Form:[Subject → A.a']B

Example: [NIS.agent → NIS.DBmember'] NIS

Third-Party Delegation: In third-party delegation, some Issuer B exercises their right to delegate a Role defined in A's namespace. A and B are not the same Entity.

Form:[Subject → A.a]B

Example: [Kim → NIS.DBmember] Bob

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. PROOFS OF DELEGATIONS WITH DRBAC IN THE SCENARIO

Step 2
(1)[Kim → ROK JCS.operations officer]ROK JCS (2)[ROK JCS.operations officer → China.officer with China.air-quality = Korea area and China.secret-level <= TS] China (3)[China.officer → China-SEPA.account]China (4)[China-SEPA.account → China-SEPA.database with China.air-quality = ‘ and China.secret-level <= ‘] China (5)[China-SEPA.database → China-SEPA.access with China.air-quality = Sin-uiju and China.secret-level <= TS] CIA
Result
[Kim → China-SEPA.access with China.air-quality = Sin-uiju and China.secret-level <= TS] CIA

Step 4
[John → UN IAEA.observer]UN [UN IAEA.observer → US AIS.member with US AIS.images = Korea area and US AIS.data = Korea area and US AIS.secret-level <= TS] US AIS [US AIS.member → US AIS.database with US AIS.images = ‘ and US AIS.data = ‘ and US AIS.secret-level <= TS] US AIS [US AIS.database → US AIS.access with US AIS.images = ‘ and US AIS.data = ‘ and US AIS.secret-level <= TS] US AIS
Result
[John → US AIS.access with US AIS.images = ‘ and US AIS.data = ‘ and US AIS.secret-level <= TS] US AIS

Step 5
[James → UN Security Council.member]UN [UN Security Council.member → NIS.agent with NIS.images = Korea area and NIS.air-quality = Korea area and NIS.secret-level <= TS] NIS [NIS.agent → NIS.database' with NIS.images = ' and NIS.air-quality = ' and NIS.secret-level >= TS] NIS [NIS.database → NIS.access' with NIS.images = Sin-uiju and NIS.air-quality = Sin-uiju and NIS.secret-level <= TS] NIS
Result
[James → NIS.access' with NIS.images = Sin-uiju and NIS.air-quality = Sin-uiju and NIS.secret-level <= TS] NIS

Step 6
[Park→ ROK Navy.operations planner]ROK Navy [ROK Navy.operations planner → NIS.agent with NIS.images = Korea area and NIS.air-quality = Korea area and NIS.secret-level <= TS] NIS [NIS.agent → NIS.database' with NIS.images = ' and NIS.air-quality = ' and NIS.secret-level >= TS] NIS [NIS.database → NIS.access' with NIS.images = Sin-uiju and NIS.air-quality = Sin-uiju and NIS.secret-level <= TS] NIS
Result
[Park → NIS.access' with NIS.images = Sin-uiju and NIS.air-quality = Sin-uiju and NIS.secret-level <= TS] NIS

Step 7
[Wang→ China Army.operations planner] China Army [China Army.operations planner → ROK Navy.member with ROK Navy.deployment = Speical Force in Sin-uiju and ROK Navy.secret-level <= TS] ROK Navy [ROK Navy.member → ROK Navy.database' with ROK Navy.deployment = ' and ROK Navy.secret-level <= TS] ROK Navy [ROK Navy.database → ROK Navy.access with ROK Navy.deployment = Special Force in Sin-uiju and ROK Navy.secret-level <= TS] ROK Navy
Result
[Wang → ROK Navy.access with ROK Navy.deployment = Special Force in Sin-uiju and ROK Navy.secret-level <= TS] ROK Navy

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Alexander, Ian. “Misuse Case Help to Elicit Nonfunctional Requirements.” *Computing and Control Engineering Journal*, vol. 14, no. 1 (February 2003): 40-45.

Alexander, I. "Misuse Cases: Use Cases with Hostile Intent." *IEEE Software* (January/February 2003): 58-66.

Basin, David, Jürgen Doser and Torsten Lodderstedt. "Model driven security: From UML models to access control infrastructures." *ACM Transactions on Software Engineering and Methodology*, vol.15 no. 1 (January 2006): 39-91.

Ferraiolo, D. F., R. Sandhu, S. Gavrila, D. R. Kuhn and R. Chandramouli. “Proposed NIST standard for role-based access control." *ACM Transactions on Information and System Security*, vol. 4, no. 3 (2001): 224-274.

Freudenthal, Eric, Tracy Pesin, Lawrence Port, Edward Keenan, and Vijay Karamcheti. "dRBAC: Distributed Role-Based Access Control for Dynamic Coalition Environments." *Proceedings of the 22nd IEEE International Conference on Distributed Computing Systems*, IEEE Vienna, Austria (2002): 411- 420.

James X. Dempsey."Moving from 'Need to Know' to 'Need to Share:' A Review of the 9-11 Commission's Recommendations." Center for Democracy & Technology, Retrieved July 2007, Available from http://www.cdt.org/testimony/20040803dempsey.shtml.

McDaniel, C. R. and M. L. Tardy, "Role-Based Access Control for Coalition Partners in Maritime Domain Awareness." M.S. thesis, Naval Postgraduate School, Monterey, CA, 2005.

Michael, James B., and Duminda Wijekera. "Secure Execution Framework for Active Coalition Partners in Maritime Domain Awareness." *The CIP Report* (January 2007): 8-9.

Pauli, J. and D. Xu. “Integating Functional and Security Requirements with Use Case Decomposition.” *Proceedings of the 11th IEEE International Conference on Engineering of Complex Computer Systems*, Palo Alto, CA (August 2006):57-66.

Sandhu, R. S. and E. J. Coyne, “Role-Based Access Control Models,” *Computer* (Feb 1996): 38-47.

Sindre, G. and A. L. Opdahl. "Capturing Security Requirements through Misuse Cases." NIK 2001, *NorskInformatikkonferanse* 2001, Retrieved July 2007, Available from http://www.nik.no/2001.

Sindre, G. and A. L. Opdahl, "Eliciting Security Requirements by Misuse Cases." *Proceedings of the 37th Conference on Techniques of Object-Oriented Languages and Systems*, IEEE, Sydney, Australia (2000): 120-131.

Sindre, G. and A. L. Opdahl. "Templates for Misuse Case Description." *Proceeding of the 7th International Workshop on Requirements Engineering*, Foundation for Software Quality, Interlaken, Switzerland (June 2001): 125-137.

U.S. Department of State, "State Sponsors of Terrorism," Retrieved August 2007, Available from http://www.state.gov/s/ct/c14151.htm.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, VA

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, CA

3. Army Headquarters Library
   The Republic of Korea Army
   Gye Ryong, Korea

4. Woo Dang Library
   Korea Military Academy
   Seoul, Korea

5. Prof. J. Bret Michael
   Naval Postgraduate School
   Monterey, CA

6. Prof. Duminda Wijesekera
   George Mason University
   Fairfax, VA

7. Mr. Christopher Newcomb
   SPAWARSYSCOM Code 05 TENCAP West
   San Diego, CA

8. Mr. Frederick Glaeser
   SPAWARSYSCOM Code 05 TENCAP West
   San Diego, CA

9. Mr. James Mueller
   Naval Postgraduate School
   Monterey, CA

10. Prof. Alan Ross
    Naval Postgraduate School
    Monterey, CA

11. Prof. Herschel Loomis
Naval Postgraduate School
Monterey, CA