Fall 2010

Cloud security requirements analysis and security policy development using a high-order object-oriented modeling technique

Kenneth Kofi Fletcher

Department:

CLOUD SECURITY REQUIREMENTS ANALYSIS AND SECURITY POLICY DEVELOPMENT USING A HIGH-ORDER OBJECT-ORIENTED MODELING TECHNIQUE

by

KENNETH KOFI FLETCHER

A THESIS

Presented to the Faculty of the Graduate School of the

MISSOURI UNIVERSITY OF SCIENCE AND TECHNOLOGY

In Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE IN COMPUTER SCIENCE

2010

Approved by

Xiaoqing (Frank) Liu, Advisor
Sanjay K. Madria
Vincent Yu (Wen-Bin)

ABSTRACT

Security continues to be a major challenge for cloud computing, and it is one that must be addressed if cloud computing is to be fully accepted. Most technological means of securing non-cloud computing systems can be either applied directly or modified to secure a cloud; however, no integrated model-based methodology is yet available to analyze cloud security requirements and develop policy to deal with both internal and external security challenges. This work proposes just such a methodology and demonstrates its application with specific cases. Cloud assets are represented by high-order object models, and misuse cases together with malactivity swimlane diagrams are developed to assess security threats hierarchically. Cloud security requirements are then specified, and policies are developed to meet them. Examples show how the methodology can be used to elicit, identify, analyze, and develop cloud security requirements and policies using a structured approach, and a case study evaluates its application by a cloud service provider. Finally, the work shows how the prevention and mitigation security policies presented here can be conveniently incorporated into the normal functionality of a cloud computing system.

ACKNOWLEDGMENTS

I am indebted to many people and wish to thank all who have helped me through the course of this project. First of all, I thank my advisor, Dr. Xiaoqing (Frank) Liu, who has encouraged and challenged me throughout the research. Secondly, I would like to thank Drs. Sanjay K. Madria and Vincent Yu for serving on my thesis committee and taking time to review this work.

I am most grateful to my parents Mr. and Mrs. Albert Fletcher, and my sister Alberta Fletcher for their encouragement and support. I also want to thank my friends Prof Richard Amankwah of the University of Mines and Technology, Ghana, Mr. and Mrs. Robert Bowers, and Mr. Andy Belval for their help and support. Finally, I want to thank Jeanine E. Bruening, editor at the Writing Center, for helping me to edit this thesis.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

Page

1. INTRODUCTION

Like all computing systems, cloud computing systems that consider security from the initial requirements and design stages are more secure than those that address security only once the system is in place. Nonetheless, security requirements are generally not analyzed early enough in the system development process, and few organizations proactively safeguard sensitive business information stored in the cloud because they lack cloud-specific security policies [15]. Due to the complexity of the cloud environment, effective testing demands that nonfunctional requirements such as those related to security be analyzed and policies be developed early to address them in the development process using a comprehensive approach that considers the entire cloud.

The unified modeling language (UML) [6] that is most often employed to elicit of requirements was not initially designed to capture nonfunctional requirements such as security requirements. As explained in Section 2 below, existing methods to analyze security requirements do not consider both internal and external threats in a structured manner. They focus entirely on external misusers and rely only on security technologies such as network monitoring systems, intrusion detection and prevention systems, firewalls, antivirus systems, and data leakage protection.

Internal threats have steadily increased over the past few years, and cloud computing is not necessarily any more secure internally than noncloud computing environments. Internal misusers generally have more knowledge of and access to data and applications than do external misusers. Although internal threats cannot be entirely eliminated, some effective barriers can be developed to mitigate them.

It is crucial, therefore, to use a top-down approach based on a clear policy to analyze security requirements and develop effective security policies. Although security policies themselves do not solve problems, and in fact can actually complicate things if they are not clearly written and consistently observed, policies do define an ideal toward which all organizational efforts should point. Therefore, a systematic methodology and process are necessary to analyze security requirements and develop security policies for cloud computing systems. This methodology must identify security requirements at multiple levels to address threats, through user scenarios, posed by both internal and external misusers and thus to develop clear cloud security policies that ensure the security of the cloud environment. The process presented here employs the high-order object-oriented modeling technique [2] together with use cases [6], misuse cases [9] and malactivity swimlane diagrams [8].

1.1. BACKGROUND OF CLOUD COMPUTING

Cloud computing has emerged in recent years as a new and important computing paradigm; it is gaining increased attention in the service computing community. According to the National Institute of Standards and Technology, the cloud computing model grants convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction [4]. Cloud computing is still evolving; therefore, its definitions, applications, underlying technologies, issues, risks, and benefits continue to be refined.

Cloud service providers (CSPs) deliver applications and services that run in the cloud; that is to say, they are accessible through the web. A key attraction of cloud computing services is that they conceal the complexity of the infrastructure from developers and end users. Hence developers and users do not know or need to know what is in the cloud – they require only that it deliver the services they need**.** CSPs offer three basic services: infrastructure as a service (IaaS), platform as a service (PaaS), and software as a service (SaaS). All these services offer scalability and multitenancy. In addition, they are self-provisioning and can be deployed through private, public, community, or hybrid cloud deployment modules.

1.2. HIERARCHICAL CLOUD ARCHITECTURE

This section presents background information on various architectural elements that form the basis for cloud computing. Figure 1.1 shows a hierarchical design of cloud computing architecture. The figure is best explained from the bottom up. At the bottom is the system level, which serves as a foundation and the backbone of the cloud. It consists of a collection of data centers that supply the computing power in the cloud environment. At this level, there exist enormous physical resources such as storage disks, CPUs, and memories.

Just above the system level is the virtualization level. Virtualization, the factor that facilitates cloud computing, is an abstraction of applications and services from the underlying physical services. It is achieved with the help of a hypervisor, a software or hardware that serves as a bridge between physical devices and virtual applications. This

abstraction ensures that no application or service is tied directly on the hardware resources. This level manages the physical resources and allows sharing of their capacity among virtual instances of servers, which can be enabled or destroyed on demand. The physical cloud resources and their virtualization capabilities form the basis for delivering IaaS.

The user-level middleware includes software-hosting platforms such as Web 2.0 Interfaces that permit developers to create rich, cost-effective user interfaces for web-based applications. It also provides the programming environments and tools that ease the creation, deployment, and execution of applications in clouds. This level aims at providing PaaS capabilities.

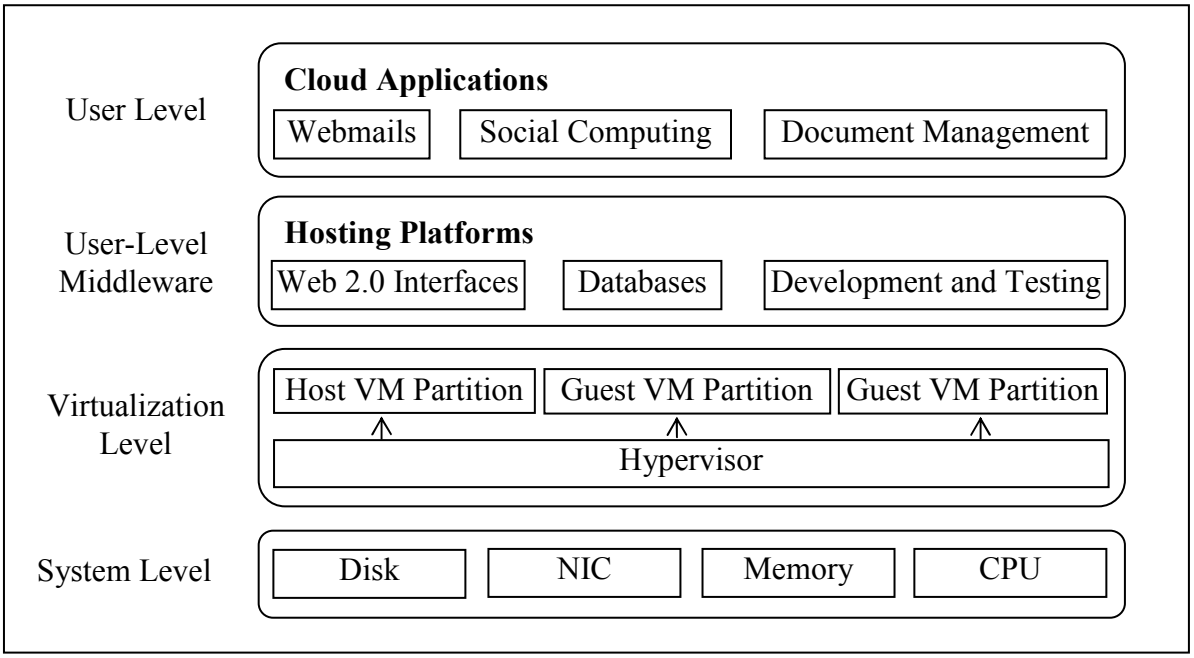

Figure 1.1. Hierarchical Cloud Computing Architecture

The top user level focuses, as its name suggests, on providing application services by making use of services provided by the lower levels. It provides SaaS capabilities. SaaS or PaaS services are often developed and provided by a third party distinct from the IaaS provider [16].

1.3. THE HIGH-ORDER OBJECT-ORIENTED MODELING TECHNIQUE (HOOMT)

The HOOMT addresses a challenge faced by requirement analysts and software engineers to develop well-structured object-oriented software systems [2]. It incorporates the object-oriented paradigm seamlessly into a structured analysis [2]. It also permits the development of object, functional, and dynamic models hierarchically according to their abstraction levels. The process eliminates incompatibility between a flat object model, in which all modeling elements are analyzed at a single level of abstraction, and hierarchical functional and dynamic models, in which modeling elements are analyzed at multiple levels of abstraction. It uses hierarchical decomposition in the analysis and design of object functionality and dynamic behavior. HOOMT also has a unique starting point and incorporates nonfunctional requirements. It has three models: the high-order object model (HOOM), the hierarchical object information flow model (HOIFM), and the hierarchical state transition model (HSTM). This work uses HOOM extensively to model the assets of the target system (i.e., the cloud) hierarchically. Liu, Lin and Dong [2] described HOOMT notation in detail.

2. RELATED WORK

Although there has been much discussion of cloud computing security concerns, few studies have focused on security policies. Hanna [1] proposed a streamlined security analysis process to capture and analyze security requirements in cloud computing. His method identifies the assets to be protected and the attacks that could be mounted against these assets. It then identifies countermeasures. The process prevents or mitigates threats posed to the cloud by external misusers; however, it gives little consideration to threats posed by internal misusers, especially those who have authorized access.

A number of proposals address security concerns early in the development lifecycle. Ware, Bowles, and Eastman [10] offer a methodology to elicit security requirements using common criteria and use cases. Their work extends existing UML use case notation used to model requirements so that it can capture actor threats. Their approach identifies potential threats by developing actor profiles and identifying threats based on relationships among actors in a use case [10]. Sindre and Opdahl [9] also extend use cases, which describe what a system should do, to misuse cases, concentrating on what should not happen in a system. Their approach combines both use-case diagrams and misuse-case diagrams in a single diagram and introduces new relationships like *prevents* and *detects*. Sindre [8] has also developed malactivity swimlane diagrams, using them to capture attacks that could complement misuse cases and thus permitting early elicitation of security requirements. His technique permits the inclusion of both hostile and legitimate activities.

3. SECURITY CHALLENGES FACED BY CLOUD COMPUTING SYSTEMS

Because its applications and services are delivered through the internet, cloud computing is prone to various kinds of external security risks such as denial-of-service (DoS) and distributed denial-of-service (DDoS) attacks. In addition, and particularly in the public cloud deployment module, since data is hosted by the CSP, trust, confidentiality, and privacy are also important issues. Finally, communication among clouds must be secured to prevent man-in-the-middle (MITM) attacks.

Although data stored in the cloud and other compute capabilities are not actually in the cloud; they reside in data centers housing hundreds of servers, thousands of networking cables, and other physical devices. Nonetheless, physical threats are among the greatest dangers to the cloud. Most CSPs are acutely aware of these threats to their core IT infrastructure from natural disasters, terrorist threats, fire, sabotage, and other phenomenon.

CSPs, especially IaaS providers, offer their customers the illusion of unlimited compute, network, and storage capacity, often coupled with a frictionless registration process that allows anyone with a valid credit card to register and begin using cloud services immediately [12]. The relative anonymity of these registration and usage models encourages spammers, malicious code authors, and other misusers, who have been able to conduct their activities with relative impunity. PaaS providers have traditionally suffered most from such attacks; however, recent evidence shows that hackers have begun to target IaaS vendors as well [12].

Researchers, however, have not generally operated on the notion that security should be built around the application, not the virtual machines; therefore, hypervisors are not sufficiently robust. Further, since communication with the hypervisor contains vital information, including account names and passwords, it must be secure. Like physical computers on a physical network, virtual machines have identical IP addresses. Nearby addresses, which are visible to users in the cloud, often share the same hardware. Thus, a misuser can determine which physical servers a victim is using within the cloud, implanting a malicious virtual machine at that location from which to launch an attack [5].

Finally, in a virtualized environment, it is relatively easy to steal an entire virtual server, along with its data, without anyone noticing. Virtual machines are encapsulated in virtual disk files that reside on a virtual host server; therefore, anyone with the right permissions can copy the disk file and access data on it.

4. THE APPROACH

4.1. FRAMEWORK OF THE STRUCTURED DEVELOPMENT OF CLOUD SECURITY POLICIES

This section describes the approach used here to analyze security requirements and develop security policies in a cloud computing environment. It involves two phases: First, cloud security requirements are analyzed. Second, cloud security policies are developed, and measures are put in place to communicate and enforce them. Figure 4.1 shows a high-level view of the approach.

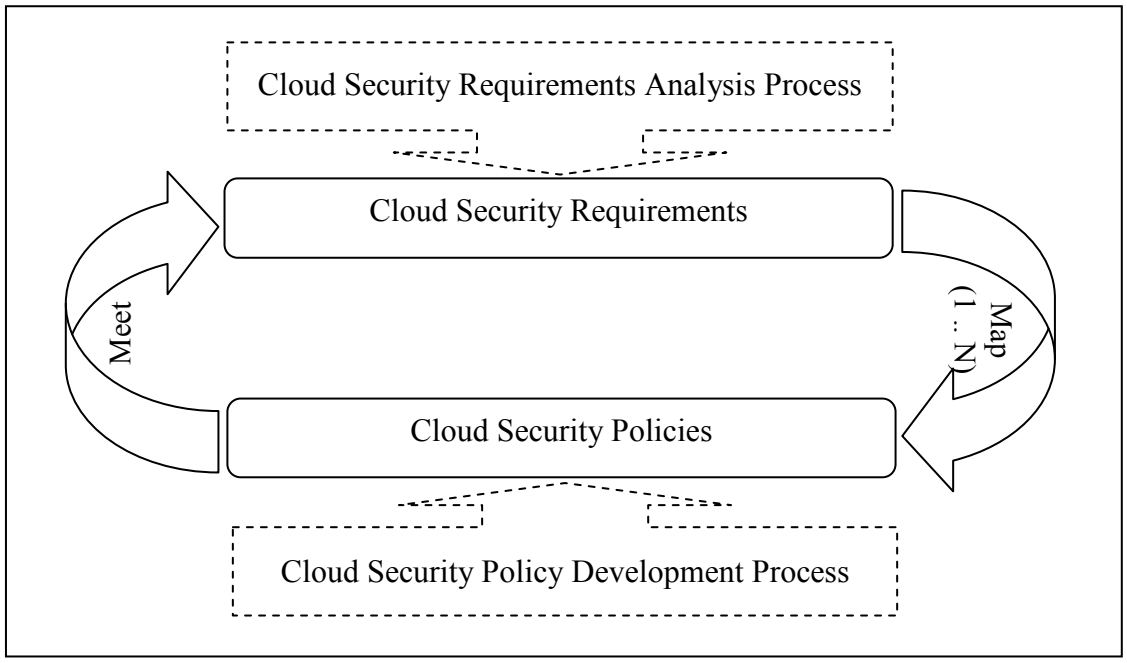

Figure 4.1. A High-Level View of the Approach

As noted above, the HOOMT, which is a major aspect of this approach to the analysis of cloud security policies, provides a structured object-oriented design

methodology based on hierarchical model development (see figure 4.2). HOOMT allows every object in the cloud to be modeled comprehensively and verified systematically for completeness. The analysis process introduced here integrates use cases, misuse cases, and malactivity swimlane diagrams with the HOOM. The malactivity swimlane diagrams decompose misuse cases, revealing in detail the activities of misusers. Also, detailed investigation of each incidence of malactivity permits development of more ways to prevent or mitigate such malactivity. This technique serves as a countermeasure for identified threats. Moreover, more threats can be identified this way; making possible the development of comprehensive cloud security policies. The result is a more efficient way to discover threats posed to cloud computing systems, both internally and externally. The structured development of the cloud security policies together with the relationships among the various diagrams at each level is shown in figure 4.2.

4.2. CLOUD SECURITY REQUIREMENTS ANALYSIS

Figure 4.3 outlines the process of analyzing to cloud security requirements. The process begins with the development of a context object diagram (COD) for the cloud computing system; this is considered as a high-order object. This COD represents the entire cloud computing system and shows its interactions with external objects such as users, either internal or external. The COD also serves as the starting point for the analysis process.

The next step is to identify use cases that describe how the cloud computing system responds to requests from users. These cases capture the behavioral requirements

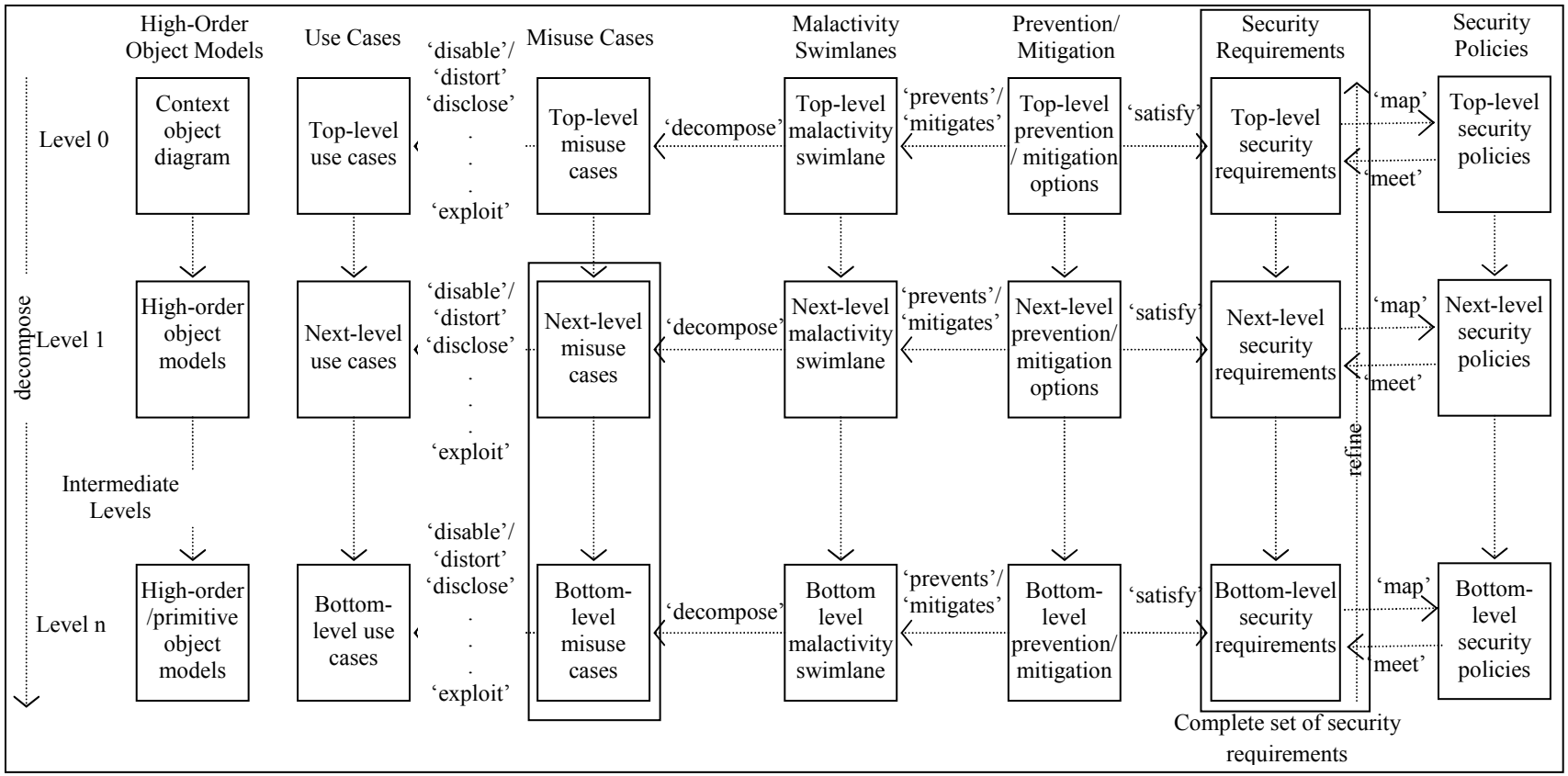

Figure 4.2. Framework of the Structured Development of Cloud Security Policies

of the cloud computing system with detailed scenarios derived from the cloud's functionalities. Next, each use case is analyzed thoroughly to determine how it could be subverted. Based on this analysis, misuse cases and misusers, either internal or external, that can harm the cloud computing system are identified. The misuse cases also reveal the various threats posed to the cloud at each level of the hierarchical model.

To identify security requirements that can serve as countermeasures to these misuse cases, the actions taken by misusers must be understood in detail. Malactivity swimlane diagrams can be used to further decompose misuse cases. Decomposition reveals the details of such misuse events and thus permits identification of more threats. It also permits the inclusion of both hostile and legitimate activities and determines the point at which prevention and mitigation options can be added to these activities to serve as countermeasures. Based on the countermeasures, security requirements are specified.

The COD is further decomposed and the cycle repeated, generating cloud security requirements at the end of every cycle. The term *decompose* refers to a process that reveals the subcomponents of the cloud object at a lower level [3]. The decomposition and security requirements analysis process continues until a stage is reached at which the cloud objects are primitive and corresponding use and misuse cases are fully explored [3]. At that point, the cloud security requirements are refined by checking for inconsistencies and ambiguities. They serve as a deliverable at the end of the first phase of the approach.

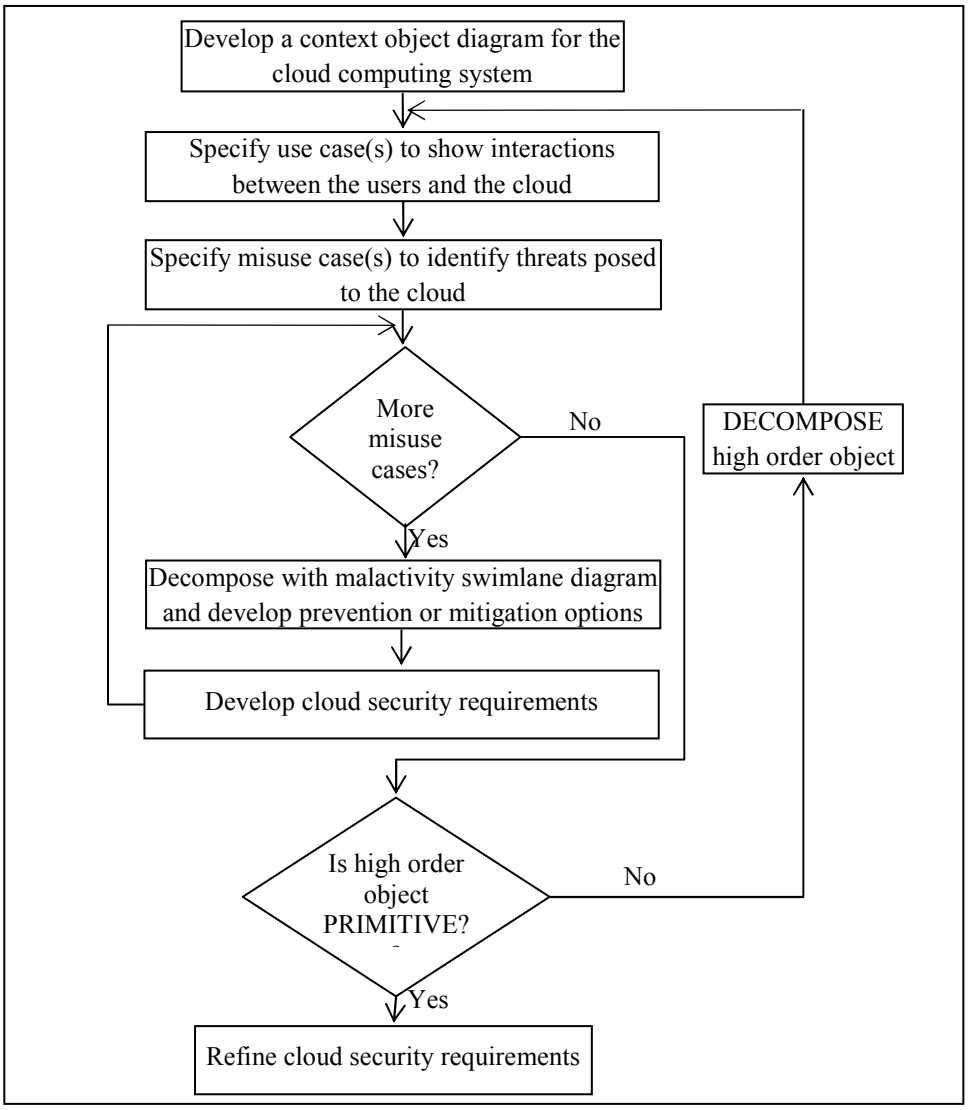

Figure 4.3. Cloud Security Requirements Process

4.3. CLOUD SECURITY POLICY DEVELOPMENT, COMMUNICATION AND ENFORCEMENT

In this work, the security policies for cloud computing systems are based on the cloud security requirements through the security requirements analysis process. Policies and requirements are not necessarily mapped one-to-one. Usually, one requirement can

be satisfied by a set of security policies. These requirements are high-level statements of countermeasures that will adequately prevent or mitigate identified misuse cases and are dependent on rigorous analysis of threats to the cloud at each level [17], as described above. Consequently, security policies are developed and integrated into the development of the cloud computing system. This approach provides a framework of best practices for CSPs and makes security policies tenable. The policies ensure that risk is minimized and that any security incidents are met with an effective response [17]. The process of developing these policies permits authorized security personnel to monitor and probe security breaches and other issues pertaining to cloud security.

The process begins with a statement articulating the motivation for developing such a policy, describing the malactivities to be governed by it, and listing the cloud assets to be protected. The problem the policy is designed to resolve is articulated. In general, the overall benefit of the policy is described. Next, those individuals or groups who must understand and observe this policy in order to perform their job are identified. Any exceptions to this policy are also noted.

At this point, the policy itself is articulated, including a description of what is actually covered by the policy, the responsibilities of the various individuals or groups involved, and the technical requirements that each individual or device must meet to comply with the policy.

Finally, once cloud security policies have been developed, they must be disseminated to users, staff, management, vendors, third party processors, and support personnel. The complexity of the cloud environment demands that some, if not all,

policies be communicated to consumers. Enforcing these policies is also an essential part of the process. This is accomplished by establishing a record that those involved have read, understood, and agreed to abide by the policies, and by discussing how violations will be handled. Figure 4.4 illustrates the process described above to develop security policies for cloud computing systems.

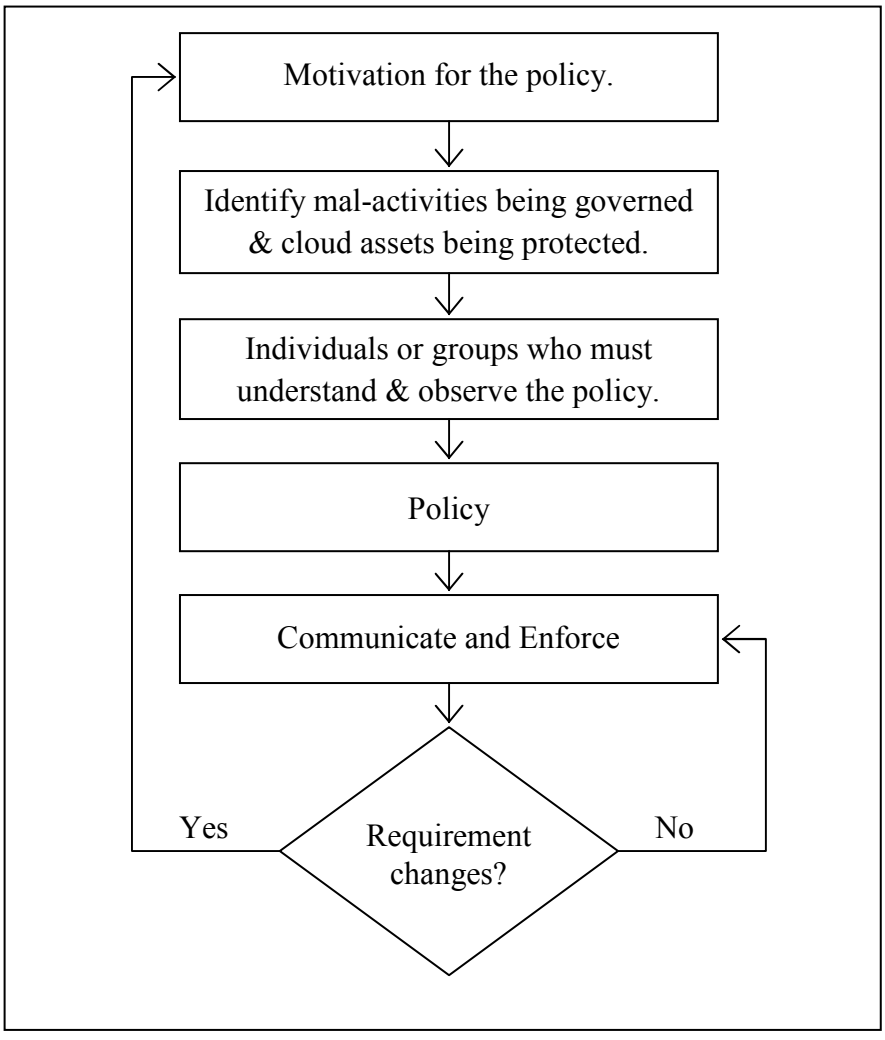

Figure 4.4. Cloud Security Policy Development Process

5. ILLUSTRATIVE EXAMPLES

The example described here illustrates how the proposed approach analyzes security requirements and develops security policies for cloud computing systems. This example involves a company that wants to create a cloud computing system to provide data hosting and processing services for the healthcare industry throughout the United States. As a CSP, this company understands the importance of secure and timely access to data for such an industry. The company also wants to maintain its own secure, state-of-the-art data center to house the servers, networking equipment, backup power systems, and other tools necessary to deliver fast, secure, and effective data services. The approach described here was used to develop a security policy document for this potential CSP.

First, the cloud was considered an object, and a COD was developed for it. The COD shows the relationship between the cloud object (i.e., the target system) and external objects including the CSP, the contingency, and the cloud end user (CEU). Natural contingencies like tornados, floods, and earthquakes can affect the availability of the cloud, as can human (intentional) actions like terrorist attacks. At this point, the cloud object is considered a high-order object; therefore, it can be decomposed into two or more high-order and or primitive objects. Figure 5.1 shows the COD of the cloud.

Next, the cloud object is decomposed to reveal its constituent objects. This represents the first level of the process, the point at which analysis of security requirements begins and the associated security policies are developed. The cloud object is decomposed into three high-order objects and one primitive object. The high-order objects are an application and related services, a hardware system, and virtualization. The

only primitive object is the service management. Decomposition of the cloud object reveals not only its constituent objects but also shows the relationships among them. See figure 5.2.

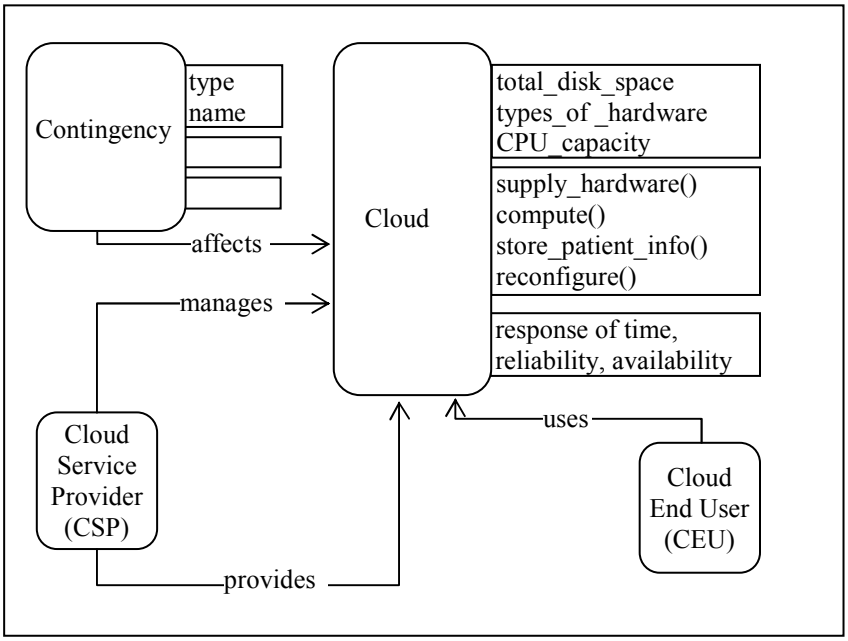

Figure 5.1. The COD of the Cloud Object

At this point, both use cases and misuse cases are specified. Figure 5.3 represents the use case-misuse case diagram of the cloud object. At this level, the misusers, whether contingency or intentional, trigger the following four misuse cases: destroy hardware system, change hardware settings, DDoS attack, and unauthorized data access. These misuse cases disable or distort the provisioning or consumption of the cloud and involve both internal and external misusers.

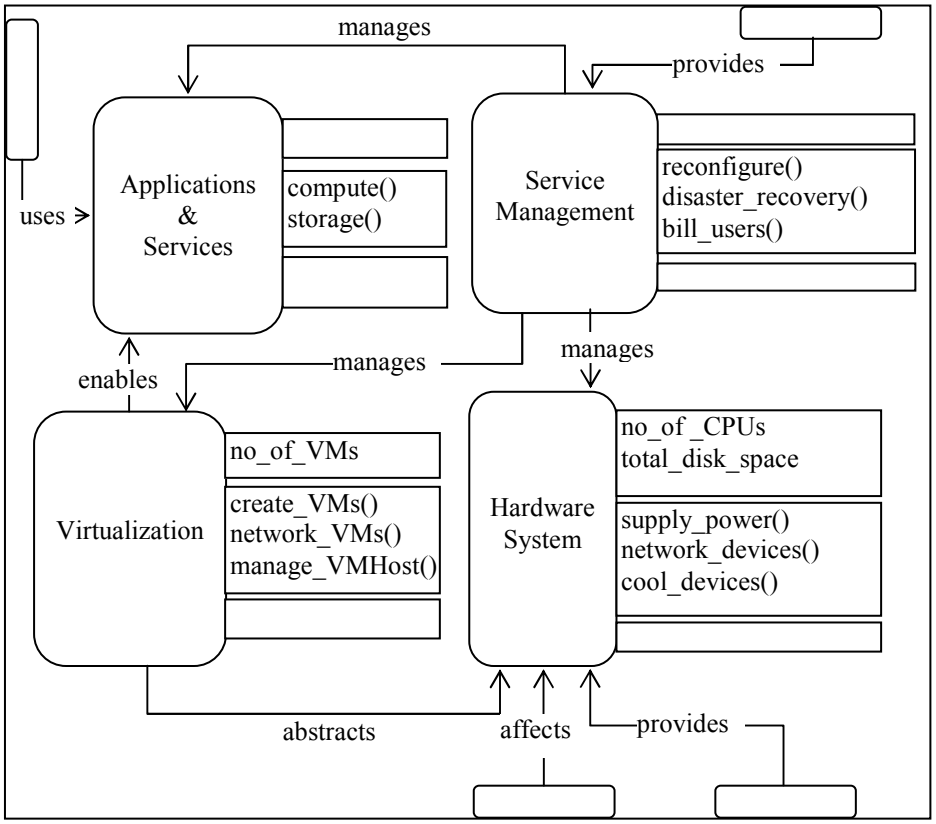

Figure 5.2. Decomposition of the Cloud Object

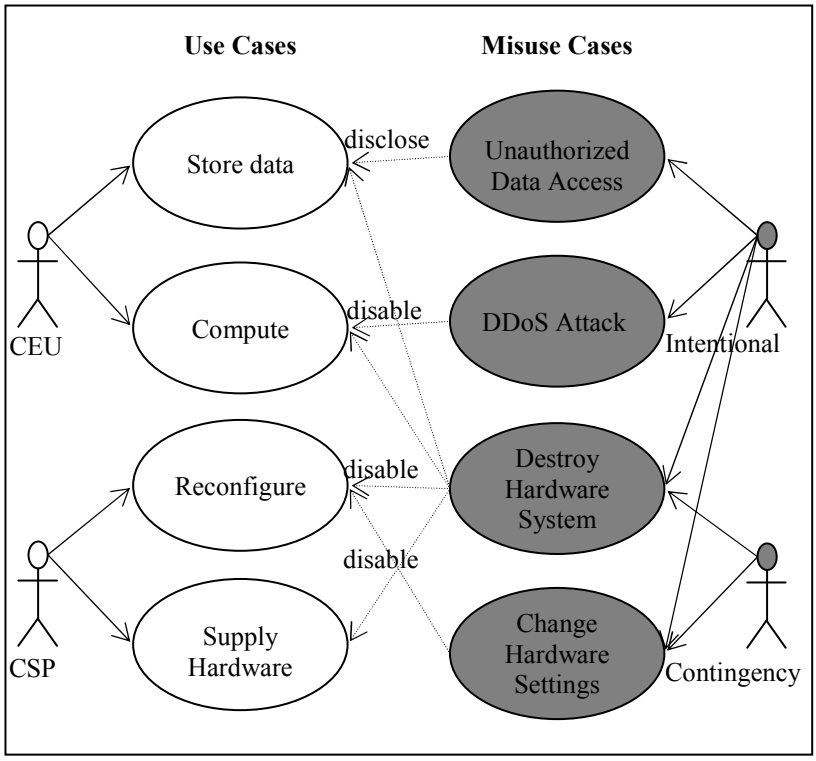

Figure 5.3. Use-Case/ Misuse Case Diagram at the Cloud Level

A malactivity swimlane diagram is developed for each misuse case to further decompose them. Sindre [8] offers a detailed description of the malactivity swimlane diagram notation. For demonstration purposes, the misuse case of unauthorized data access (an internal threat) will be decomposed at this level (see figure 5.4). In this scenario, an unauthorized user (e.g., a member of the cleaning staff) who has stolen an authorized user's badge enters the network operations center (NOC). A security staff member allows the cleaning staff access to the NOC. This security staff member is also considered a misuser because he is assumed to have connived with the cleaning staff member in this operation. The cleaning staff member then guesses the login information and retrieves confidential patient information.

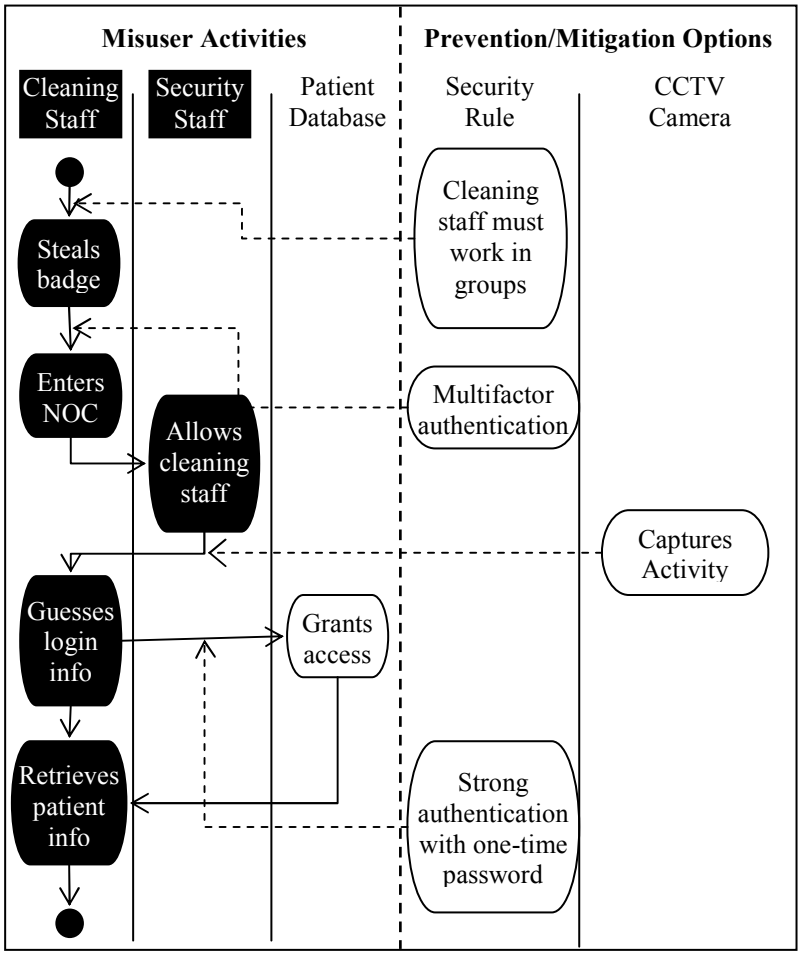

Figure 5.4. Malactivity Swimlane Diagram for the Unauthorized Data Access Misuse Case together with the Prevention or Mitigation Options

As shown in figure 5.4, decomposing misuse cases with malactivity swimlanes reveals the details of activities performed by the misuser. Thus, it is possible to determine the point in the process at which mitigation or prevention can be added. For instance, in order to prevent guesses of login details, the CSP can implement a strong authentication system with one-time password rather than just a username and password authentication.

Such an authentication method uses information sent in a short messaging service (SMS) to the user as part of the login process.

Once all misuse cases are decomposed and their respective mitigation and prevention options specified, security requirements are also developed. Figure 5.5 shows the top-level security requirements for the cloud object.

Top-Level Cloud Security Requirements (CSR):

CSR 1.1: The system must provide physical protection to all physical hardware.

CSR 1.2: The system must employ multifactor authentication with a one-time password for CEUs to prevent intrusion.

CSR 1.3: The system must monitor network requests so that any kind of distributed denial of service (DDoS) attack can be detected.

CSR 1.4: The system must audit and log CEUs, recording who logs in, when, and from where in order to recover from a breach.

CSR 1.5: The system must encrypt data in transit in order to prevent vital data from reaching unauthorized users.

Figure 5.5. Security Requirements at the Cloud-Object Level

With these security requirements, it should now be possible to determine what kind of security policies must be developed. This is done such that for every security requirement is met by at least one associated security policy. Figure 5.6 shows an example of a security policy that meets CSR 1.5.

Data-in-Transit Encryption Policy

1.0 Purpose
This document describes the encryption of data in transit to ensure the information security of the cloud. Encryption is designed to prevent unauthorized disclosure of vital information.

2.0 Scope
This policy applies to any data in transit.

3.0 Policy
All data in transit must be encrypted, and such data must be protected to prevent their unauthorized disclosure and subsequent fraudulent use.

4.0 Enforcement
Any employee found to have violated this policy may be subject to disciplinary action, up to and including termination of employment.

5.0 Definitions
Data in transit refers to any data transferred in the cloud.

6.0 Revision History
09/24/2010 - 1.0 initial policy version, Kenneth Fletcher

Figure 5.6. Security Policy to Meets CSR 1.5

The cloud security requirement analysis and policy development process then continues at the second level. The virtualization object is of particular interest in this research since virtualization is the main driver of cloud applications and services. Figure 5.7 shows the decomposition of this object into four primitive objects, and the relationships among them. The objects are the hypervisor, the virtual network system, the resource management system, and the virtual machine (VM).

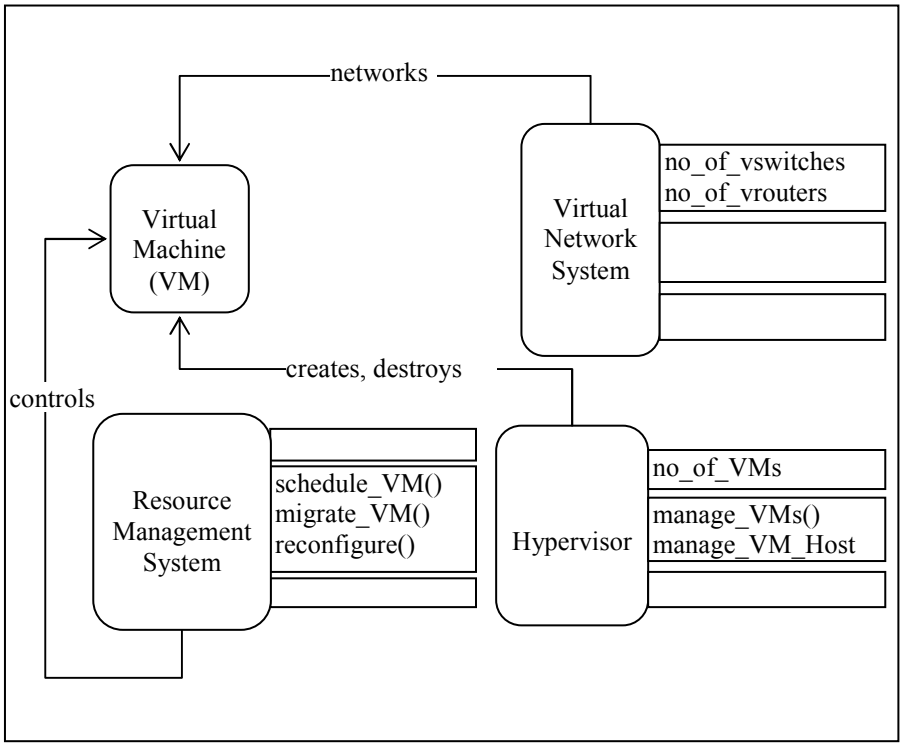

Figure 5.7. Decomposition of the Virtualization Object

Also at this level, use cases and misuse cases are specified. Four use cases were identified: create VM, reconfigure, network VMs, and manage VMs. The misusers, either contingency or intentional, initiate *plant malicious VM*, *VM escape*, and *change VM settings* misuse cases. Figure 5.8 shows the use case-misuse case diagram. At this level, the VM escape misuse case is further decomposed with the malactivity swimlane diagram. Security prevention and mitigation options are specified in the decomposition as shown in figure 5.9. *Plant malicious VM*, and *change VM settings* misuse cases are also decomposed and their respective security prevention and or mitigation options specified. Security requirements are also specified; these are shown in figure 5.10. Once these

requirements are obtained, security policies are developed to meet them. For example, figure 5.11 shows a cloud security policy developed to meet CSR 2.1.

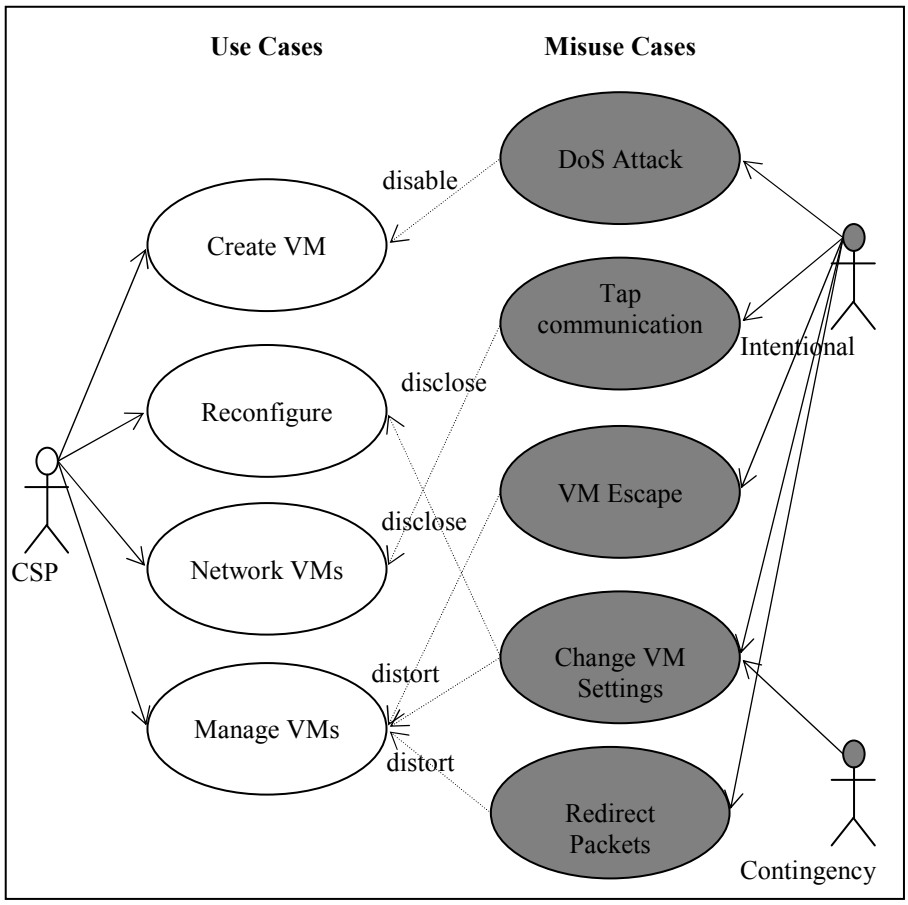

Figure 5.8. Use-Case/ Misuse-Case for the Virtualization Object

Securing the hardware system is just as important as securing the virtualization system. At the third level, the hardware system object is analyzed and further decomposed into the following four primitive objects: hardware devices, network management system, cooling system, and power system. Figure 5.12 shows the high-order object model for the decomposed hardware system object decomposed and the relationship among its primitive objects.

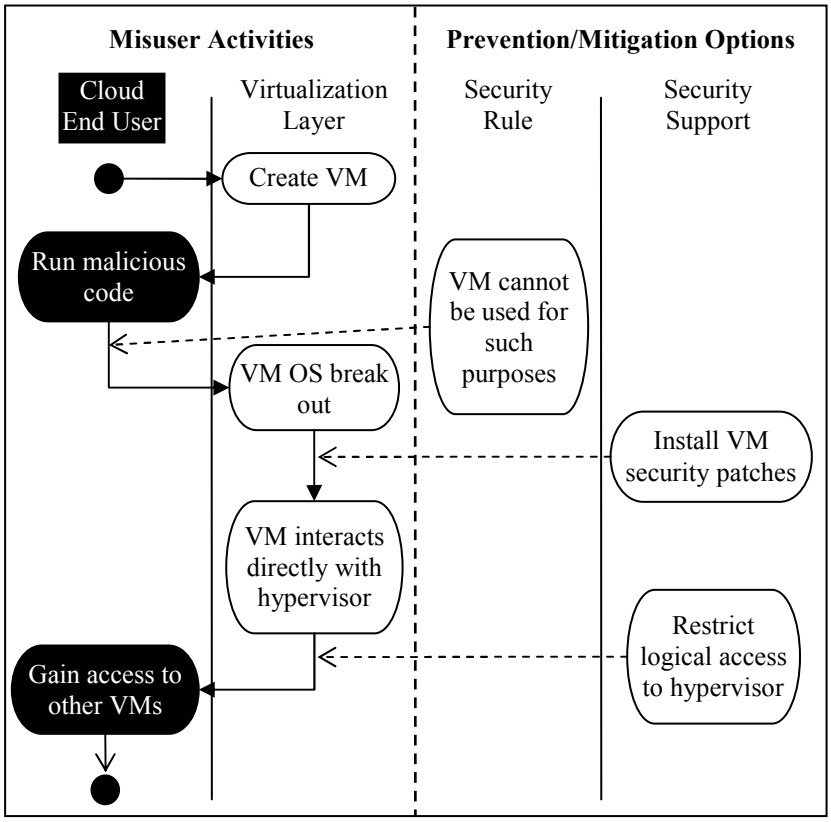

Figure 5.9. Malactivity Swimlane Diagram for the VM Escape Misuse Case together with the Prevention or Mitigation Options

Second-Level Cloud Security Requirements (CSR):

CSR 2.1: The system must restrict physical and logical access to the hypervisor to prevent VM from having direct interaction with hypervisor.

CSR 2.2: The system must employ efficient load balancing techniques to prevent VMs from causing denial of service (DoS) attacks.

CSR 2.3: The system must implement authentication of network flow such that a guest VM cannot monitor other VMs.

CSR 2.4: The system must monitor guest-host VM interaction for improper configuration changes, and in the event of any such incident it should report to the network manager.

Figure 5.10. Security Requirements at the Virtualization-Object Level

Hypervisor Access Policy

1.0 Purpose
This document describes cloud information security's required encryption of data in transit. This is designed to prevent unauthorized disclosure of vital information.

2.0 Scope
This policy applies to all nonhost virtual machines in the cloud.

3.0 Policy
All data in transit must be encrypted, and data covered by this policy must be protected to prevent their unauthorized disclosure and subsequent fraudulent use.

4.0 Enforcement
Any employee found to have violated this policy may be subject to disciplinary action, up to and including termination of employment.

5.0 Definitions
Data in transit – Data transferred in the cloud.

6.0 Revision History
09/24/2010 - 1.0 initial policy version, Kenneth Fletcher

Figure 5.11. Security Policy to Meet CSR 2.1

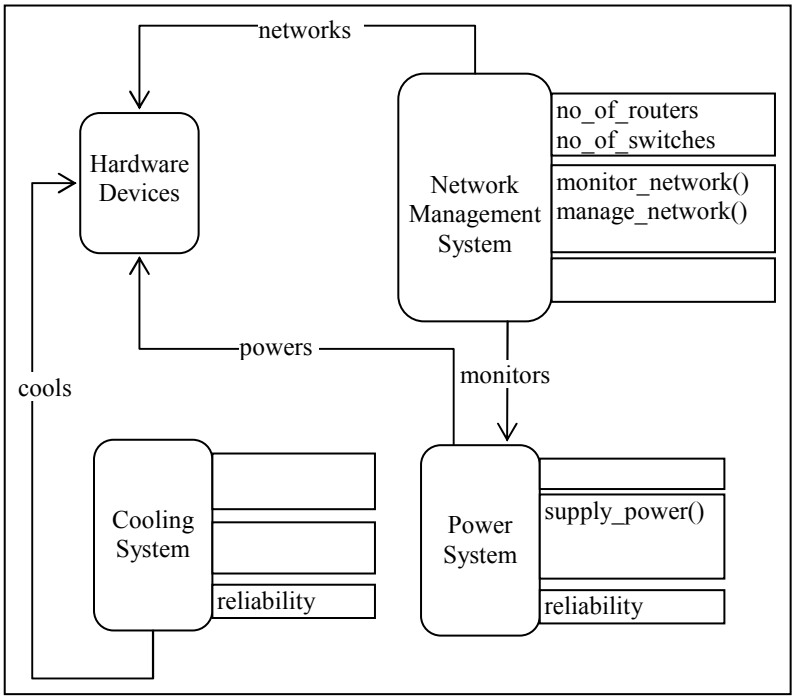

Figure 5.12. Decomposition of the Hardware System Object

Supply power, network hardware devices, and supply hardware are the three use cases specified at Level 3. Threats at this level of the cloud are mostly physical. The misusers, either contingency or intentional, initiate five misuse cases, including destroy network devices, destroy power devices, change power configuration, destroy cooling systems, and change temperature configuration. Figure 5.13 is a use-case/ misuse-case diagram for this level.

Here, the destroy power devices misuse case is decomposed further with a malactivity swimlane diagram. Figure 5.14 shows this diagram together with the security prevention options that are specified to prevent or mitigate such threats. The security requirements that are developed at this level are shown in figure 5.15. Finally, the security policy designed to meet CSR 3.2 is shown in figure 5.16.

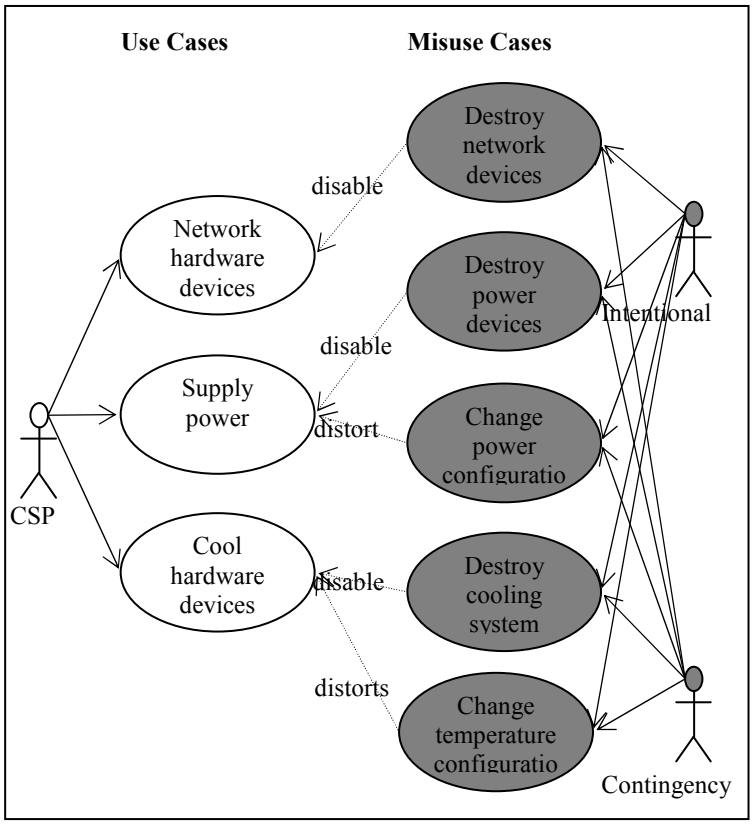

Figure 5.13. Use Case-Misuse Case for the Hardware System Object

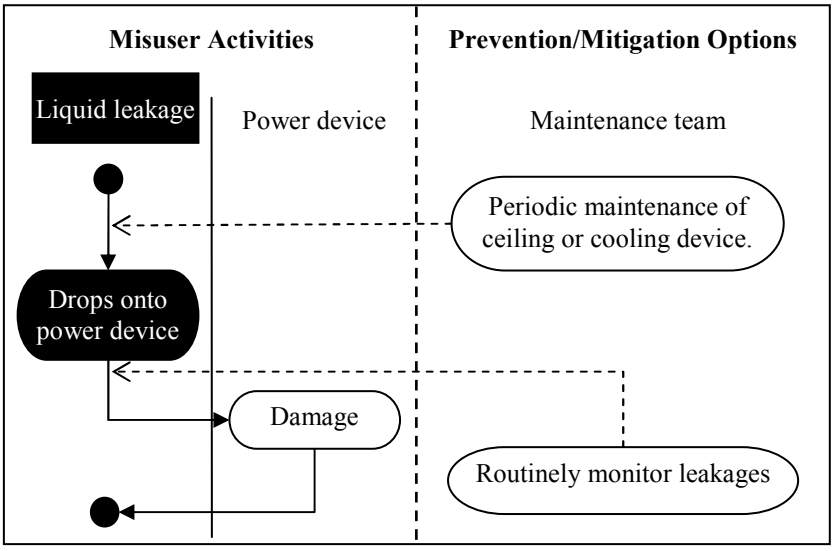

Figure 5.14. Malactivity Swimlane Diagram for the Destroy-Power-Devices Misuse Case together with the Prevention or Mitigation Options

Third-Level Security Requirements:

CSR 3.1: The system must routinely monitor power quality and load in order to detect any change in power configuration.

CSR 3.2: The system must routinely monitor temperature in order to detect any change in temperature configuration and maintain constant cooling of hardware devices.

CSR 3.3: The system must routinely monitor and detect coolant or water leaks in order to prevent destruction of power devices, cooling systems, cables, and other hardware.

Figure 5.15. Security Requirements at the Hardware System Level

Physical Devices Monitoring Policy

1.0 Purpose
This policy is intended to protect the CSP against loss of service by providing constant monitoring of hardware devices.

2.0 Scope
This policy applies to all physical devices, including but not limited to power and cooling devices.

3.0 Policy

All hardware devices shall be checked manually on a daily basis and recorded.

4.0 Enforcement
Any employee found to have violated this policy may be subject to disciplinary action, up to and including termination of employment.

5.0 Revision History
09/24/2010 - 1.0 initial policy version, Kenneth Fletcher

Figure 5.16. Security Policy to Meet CSRs 3.1

6. CASE STUDY

Section 5, explained how cloud security requirements can be analyzed and security policies developed. Here, the proposed approach is applied to a real case study involving a cloud service provider whose name has been omitted due to confidential reasons. The objective is to analyze the company's current security state and provide advice on strengthening the security of its cloud.

The company, headquartered in St. Louis, Missouri, offers highly available business solutions, including colocation, cloud computing, managed services, and insourcing in a carrier-class data center facility. The private cloud computing environment provides access to resources from storage, virtual servers, and desktops to email and mobile devices, all on an as-needed basis. These systems are powered from their own platform supported by a 30,000-square-foot state-of-the-art data center in St. Louis. In order to provide a geographically diverse redundancy system as a backup for the primary data center, the service provider operates another data center in Cincinnati, Ohio.

The private cloud offerings of the company fall primarily in the IaaS space, although it offers a number of applications that are delivered and consumed by clients on a variable per-use basis.

The security requirements analysis process begins by developing the COD of the private cloud. Because St. Louis, Missouri, is an earthquake zone, the cloud is vulnerable to natural contingencies. Figure 6.1 shows the COD and the relationships among the objects.

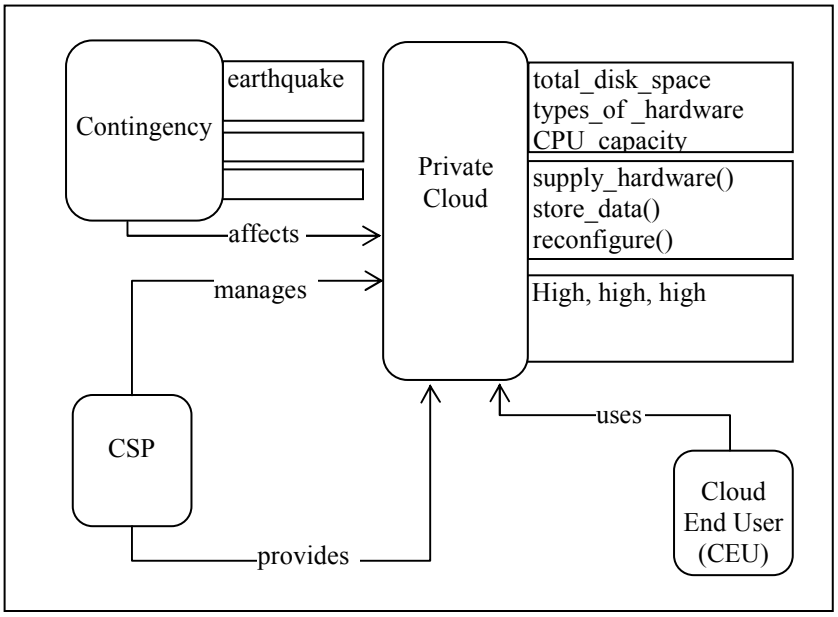

Figure 6.1. COD of the Private Cloud Object

The private cloud object is decomposed into four subobjects: three high-order objects and a primitive object. The high-order objects are services, hardware resources, and the VMware vSphere. The only primitive object is service management. Decomposition of the cloud object reveals not only its constituent objects but also the relationship among them (see figure 6.2).

Figure 6.3 represents the use-case/ misuse-case diagram of the private cloud at the cloud level. The following three misuse cases were identified: destroy hardware, change hardware settings, and unauthorized data access. These misuse cases subvert the supply hardware, reconfigure system, and store data use cases, respectively.

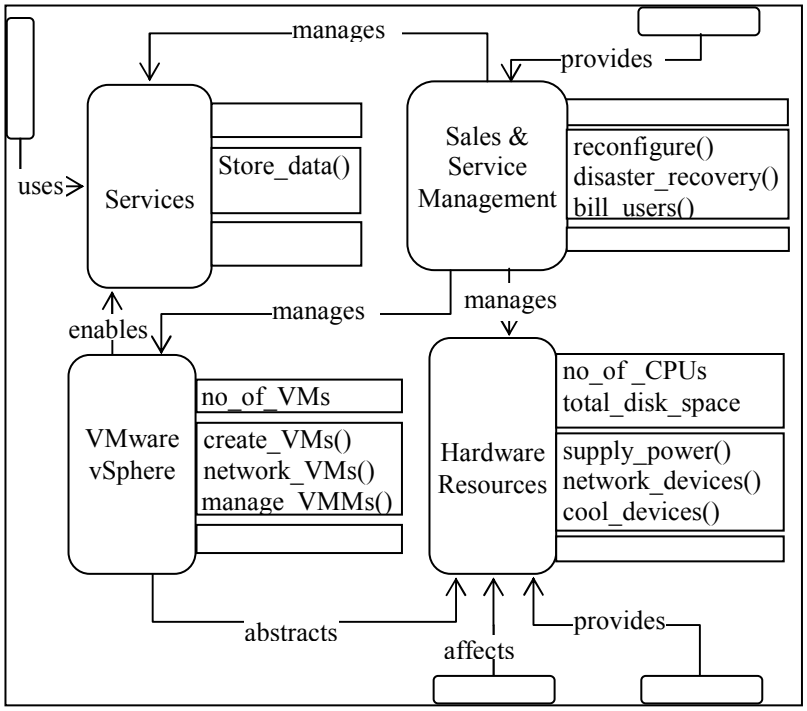

Figure 6.2. Decomposition of the Private Cloud Object

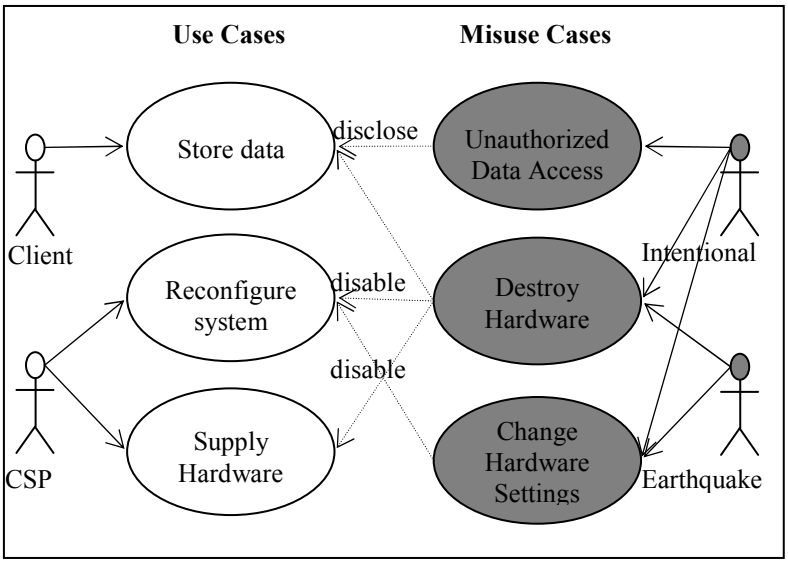

Figure 6.3. Use-Case/ Misuse-Case for the Private Cloud Object

The cloud service provider for this case study understands its datacenter as belonging to its clients and therefore permits clients access to it. With this setup, internal threats are likely to be the main security issue at this level. Therefore, the unauthorized data access misuse case is decomposed here to find determine how such a setup could be compromised and develop prevention or mitigation options to serve as countermeasures.

Figure 6.4 is a malactivity swimlane diagram describing a scenario in which one client (client A) goes into the data center to steal another client's (client B) hard drive and access confidential data on it. The decomposition clarifies the activites of client A and makes it easier to prevent them. The prevention or mitigation options specified in figure 6.4 are translated into security requirements, which are shown in figure 6.5. These are the security requirements specified at the first level of the private cloud. Figure 6.6 shows a security policy to meet CSR 1.2.

At the second level of the security requirements analysis process, the VMware vSphere object (the virtualization layer) is analyzed. Figure 6.7 shows its decomposition and the relationship existing among its four constituent primitive objects. The objects are VMware ESXi hypervisor, vCenter server, virtual machine, and application services.

Also at this level, use cases and misuse cases are specified. Three use cases were identified: create VM, vstorage, and manage VM host. The intentional misuser, whether a cloud user or the CSP itself, initiates DoS attack, VM escape, MITM attack, and redirect packets misuse cases. Figure 6.8 shows the use-case/ misuse-case diagram. At this level, the MITM attack misuse case was further decomposed with a malactivity swimlane diagram.

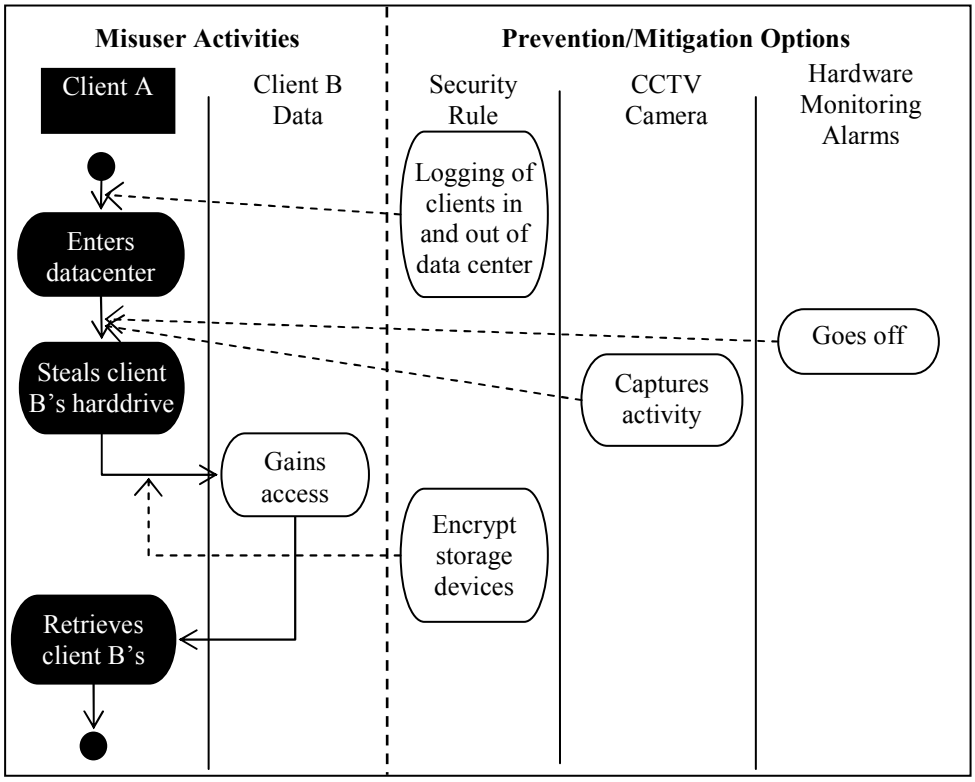

Figure 6.4. Malactivity Swimlane Diagram for the Unauthorized Data Access Misuse Case together with the Prevention or Mitigation Options

Private Cloud Object-Level Cloud Security Requirements (CSR):

CSR 1.1: The system must provide hardware monitoring alarms for all physical hardware.

CSR 1.2: The system must audit and log client and visitor access to the data center, recording who logs in and when in order to recover from a breach.

CSR 1.3: The system must encrypt data at rest in order to prevent vital data from reaching unauthorized users.

Figure 6.5. Security Requirements at the Private Cloud Level

Customer and Visitor Data Center Access Policy

1.0 Purpose
The purpose of this document is to provide guidance for customers and visitors to the data center, as well as for employees sponsoring visitors.

2.0 Scope
This policy applies to all customers and visitors to the data center and to employees who sponsor visitors.

3.0 Policy

3.1 Check-In
All visitors must arrive at a designated check-in entrance (i.e., the main reception desk) and present government-issue photo identification at time of check-in.
All visitors must be met by their employee sponsor at the time of check-in. Visitors must sign a "Visitor Agreement." All visitor electronics will be checked in as well.

3.2 Badges
Customer and visitor badges must be worn at all times. Employees are instructed to immediately report anyone not wearing a customer, visitor, or employee badge.
Visitors requiring access to areas controlled by swipe card access locks should be assisted by their sponsoring employee.

3.3 Photographs and Cameras
Customers and visitors are not permitted to take photographs inside the data center, without specific prior arrangement with sponsoring employees.

3.4 Check-Out
Visitors will check out at the same station where they arrived. All visitor electronics will be checked out.

3.5 Exit Inspection
Visitors may be subject to a brief search of their laptop bags or other luggage as they exit the data center.

4.0 Enforcement
Violation of any part of this policy by any employee will result in suitable disciplinary action, up to and including prosecution and or termination.
Violation of any part of this policy by any visitor can result in similar disciplinary action against the sponsoring employee, and can also result in termination of services or prosecution in the case of criminal activity.

6.0 Revision History
09/24/2010 - 1.0 initial policy version, Kenneth Fletcher

Figure 6.6. Security Policy to Meet CSR 1.2.

Figure 6.9 shows this decomposition together with the prevention or mitigation options. The standard MITM ARP cache spoofing attack is still an issue with the VMware vSpehre object. This attack occurs when a victim thinks the attacker is the default gateway and the actual default gateway thinks otherwise.

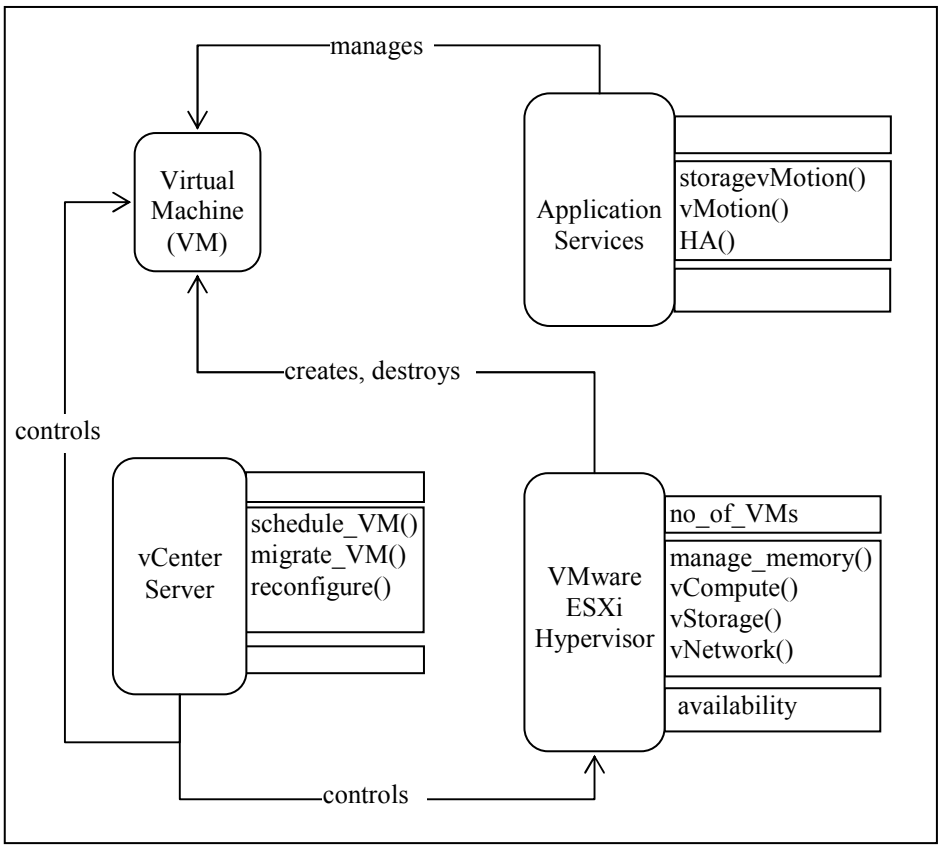

Figure 6.7. Decomposition of the VMware vSphere Object

During the course of this attack the victim sends packets to the attacker (default gateway) who then copies the information, stops it, or at worst changes the contents of the frame itself. The modified or copied frame is sent to the unsuspecting default gateway

(actual) for further processing. When the receiving packet returns, the data can be similarly intercepted.

Other misuse cases identified here were also decomposed and their respective security prevention or mitigation options specified. Security requirements for this level are shown in Figure 6.10. Finally at this level, security policy to meet CSR 2.1 is developed as shown in figure 6.11.

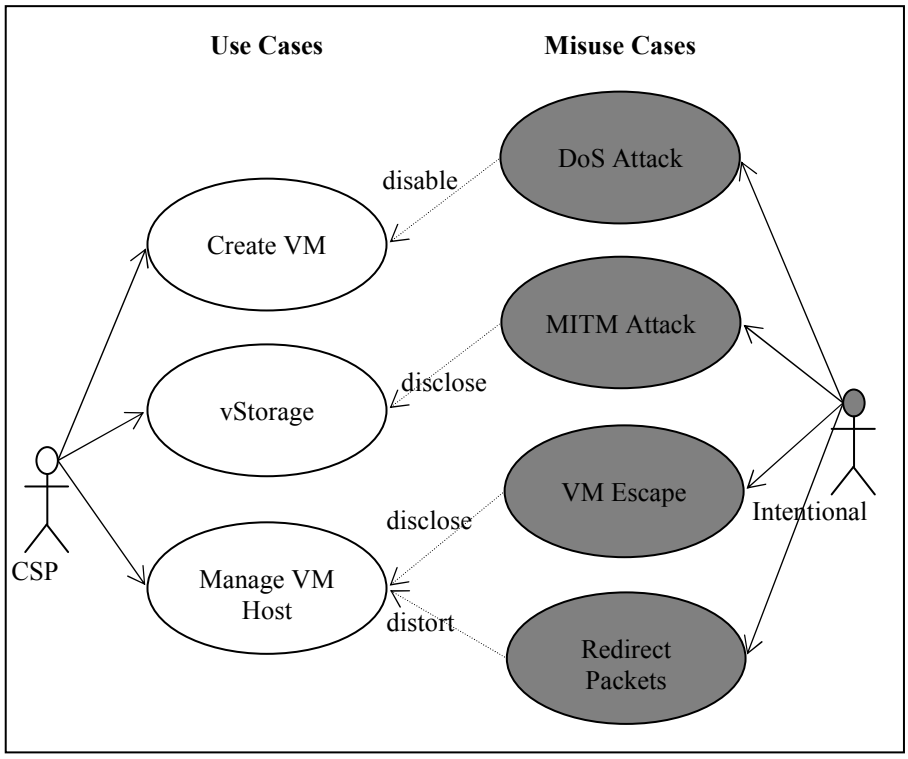

Figure 6.8. Use-Case/ Misuse-Case for VMware vSphere Object

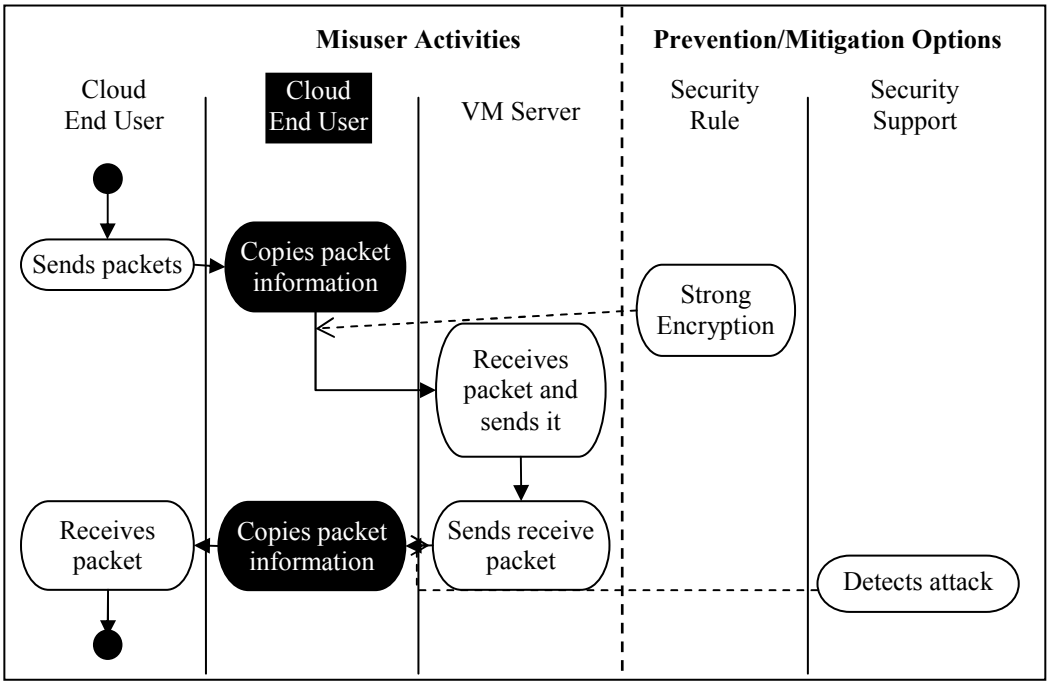

Figure 6.9. Malactivity Swimlane Diagram for the MITM Attack Misuse Case together with Prevention or Mitigation Options

VMware vSphere Object-Level Cloud Security Requirements (CSR):

CSR 2.1: The system must isolate all traffic to and from storage repositories (data-in-motion) from other nonstorage traffic.

CSR 2.2: The system must not allow VM repositories or datastores to be accessible to other VMs except for the VM host servers.

CSR 2.3: The system must restrict physical and logical access to the hypervisor to prevent VM from having direct interaction with hypervisor.

CSR 2.4: The system must employ efficient load balancing techniques to prevent VMs from causing denial of service (DoS) attacks.

Figure 6.10. Cloud Security Requirements at the VMware vSphere Object Level

Data-in-Motion Isolation Policy

1.0 Purpose
The purpose of this policy is to meet CSR 2.1, which defines the isolation of data in motion in order to prevent MITM attacks when using and managing virtualization with VMware vSphere technologies.

2.0 Scope
This policy applies to any data-in-motion

3.0 Policy
All data in transit must be isolated by employing storage area network (SAN) connectivity, that is, a network of servers and storage devices independent of the ethernet network.

4.0 Enforcement
Any employee found to have violated this policy may be subject to disciplinary action, up to and including termination of employment.

5.0 Definitions
Data in transit refers to data transferred in the VMware vSphere vStorage medium.

6.0 Revision History
09/24/2010 - 1.0 initial policy version, Kenneth Fletcher

Figure 6.11. Security Policy to Meet CSR 2.1.

Finally, the hardware resources object was analyzed and further decomposed into the following four primitive objects: hardware devices, network management system, cooling system, and power system. The high-order object model for the hardware resources object and the relationship between its primitive objects are represented in figure 6.12.

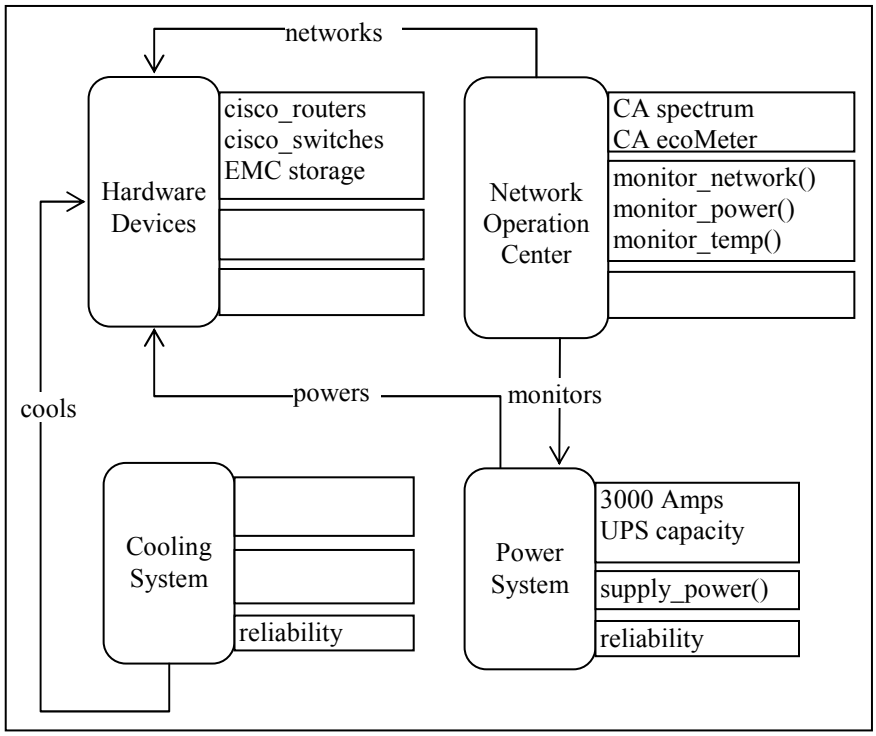

Figure 6.12. Decomposition of the Hardware Resources Object

Supply power, network hardware devices, and supply hardware are three use cases specified at the hardware resources level. Threats at this level of the private cloud are mostly physical. The misusers, contingency and intentional, initiate five misuse cases, including destroy network devices, destroy power devices, change power configuration, destroy cooling systems, and change temperature configuration. Figure 6.13 shows the use-case/ misuse-case diagram for this level.

Here, the destroy power devices misuse case was decomposed further with a malactivity swimlane diagram. Figure 6.14 shows this decomposition together with the security prevention or mitigation options for this threat. Figure 6.15 shows the security

requirements that were developed at level 3. Also, figure 6.16 shows the security policy developed to meet CSR 3.1.

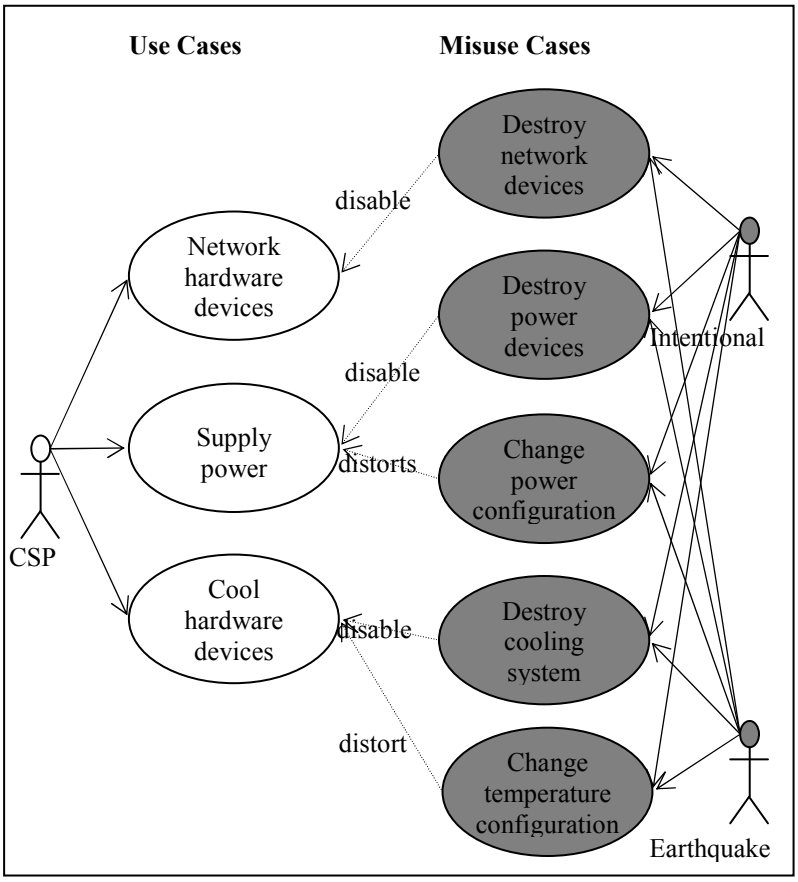

Figure 6.13. Use-Case/ Misuse-Case for the Hardware Resources Object

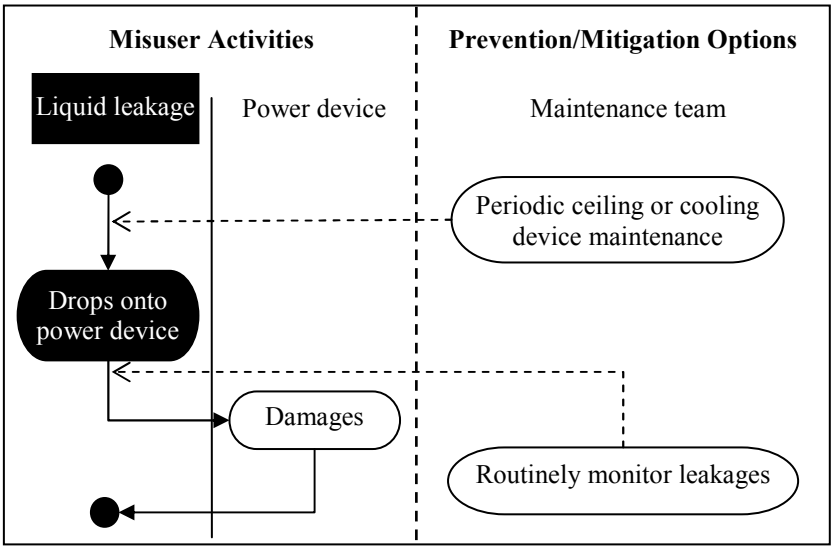

Figure 6.14. Malactivity Swimlane Diagram for the Destroy Power Devices Misuse Case together with the Prevention or Mitigation Options

Hardware Resources Object-Level Cloud Security Requirements (CSR):

CSR 3.1: The system must routinely monitor power quality and load in order to detect any change in power configuration.

CSR 3.2: The system must routinely monitor temperature in order to detect any change in temperature configuration and maintain constant cooling of hardware devices.

CSR 3.3: The system must routinely monitor and detect coolant or water leaks in order to prevent destruction of power devices, cooling systems, cables, and other hardware.

Figure 6.15. Security Requirements at the Hardware Resources Object Level

Periodic Maintenance Policy

1.0 Purpose
The purpose of this document is to define the standards for effective maintenance of the private cloud's assets so that equipment remains safe at all times.

2.0 Scope
This policy applies to all equipment serving the CSP's data center.

3.0 Policy

3.1 Maintenance Standards
Each piece of equipment will be allocated an importance rating of 1 - 5. Maintenance standards will vary depending on the importance of the facility, per the guide below:

1. Not important: Carry out only essential maintenance.
2. Low importance: Defer non-essential maintenance where possible.
3. Fair importance: Carry out maintenance based on risk assessment.
4. Important: Maintain to the best standard that resources allow.
5. Very important: Maintain to a very high standard.

3.2 Maintenance Categories
Each equipment must be categorized as one of the following: preventive maintenance, statutory maintenance, corrective maintenance, or backlog maintenance.

4.0 Enforcement
Violation of any part of this policy by any employee will result in suitable disciplinary action, up to and including prosecution and or termination.

5.0 Revision History
09/24/2010 - 1.0 initial policy version, Kenneth Fletcher

Figure 6.16. Security Policy to Meet CSR 3.2.

7. CONCLUSIONS AND FUTURE WORK

Cloud computing is becoming popular and represents the future of computing. Before it can be embraced by individuals and enterprises, however, the issue of security must be addressed. Early consideration of security in cloud computing systems places it on a par with other functional requirements of the system and significantly improves the security of the system. This work has successfully addressed these security issues, by developing a process to determine security requirements and develop policies for a cloud computing system level-by-level in a structured manner. This methodology analyzes security requirements by identifying threats posed by misusers both external and internal to a system. The process was applied here to typical cloud architecture to demonstrate its function and it was further applied to an actual case study of a cloud service provider in St. Louis, Missouri. In each case, misuse cases at three different were identified. Malactivity swimlane diagrams for these misuse cases were generated, permitting development of countermeasures for prevention or mitigation. Security requirements were then derived based on the prevention or mitigation options. Finally, security policies were developed to meet at least each requirement.

Developing comprehensive cloud-specific security policies is a very difficult task that requires collaboration and insight from many individuals in various areas of expertise. This is very important because the cloud architecture is very broad. If not written clearly and consistenly observed, cloud security policies can actually complicate things rather than helping to prevent or mitigate security issues in cloud computing. Enforcing cloud security policies are difficult and require management, employee, and

user support. Also, due to the difficulty that enforcing secuirty policies bring, it is harder to evaluate security compliance in cloud computing. For future work, I will research into and develop cloud-specific security metrics in order to quantify security in cloud computing to find out how safe the cloud is from time to time.

BIBLIOGRAPHY

[1] S. Hanna, “Cloud Computing: Finding the silver lining,” [powerpoint slides]http://www.ists.dartmouth.edu/docs/HannaCloudComputingv2.pdf, March 2009

[2] X. F. Liu, H. Lin, and L. Dong, “High-Order Object-Oriented Modeling Technique for Structured Object-Oriented Analysis,” International Journal of Computer and Information Science (IJCIS), 3rd ed., vol. 2. Oxford: Clarendon, 2001, pp.68–73.

[3] S. Markose, X. F. Liu, and B. McMillin, “A Systematic Framework for Structured Object-Oriented Security Requirements Analysis in Embedded Systems,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 2008, pp. 271–350.

[4] National Institute of Standards and Technology. “Cloud computing,” http://csrc.nist.gov/groups/SNS/cloud-computing/index.html, October 2009.

[5] T. Ristenpart, E. Tromer, H. Shacham, and S. Savage, “Hey, You, Get Off of My Cloud: Exploring Information Leakage in Third-Party Compute Clouds,” Proc. 16th ACM Conference on Computer and Communications Security, 2009, pages 199–212

[6] J. Rumbaugh, I. Jacobson, and G. Booch, The Unified Modeling Language Reference Manual. Addison-Wesley, Erewhon, NC, 1999.

[7] M. Ryan, S. Markose, Y. Cheng, X. F. Liu, and B. McMillin, "Structured Object-Oriented Co-analysis/Co-design of Hardware/Software for the FACTS Power System" Proc.29th IEEE Annual International Computers Software and Applications Conference, 2005.

[8] G. Sindre, "Malactivity Diagrams for Capturing Attacks on Business Processes" Proc.13th International Working Conference, REFSQ., 2007, pages 355–366,.

[9] G. Sindre and A. L. Opdahl, "Eliciting Security Requirements with Misuse Cases," Science, vol. 294, Dec. 2001, pp. 2127-2130, doi:10.1126/ science.1065467.

[10] M. S. Ware, J. B. Bowles, and C. M. Eastman, "Using the Common Criteria to Elicit Security Requirements with Use Cases." Proc. IEEE Symp. Computational Intelligence in Scheduling (SCIS 07), IEEE Press, Dec. 2007, pp. 57-64, doi:10.1109/SCIS.2007.357670.

[11] Cloud Security Alliance, "Top Threats to Cloud Computing V 1.0" http://www.csa.org/pdf, March 2010

[12] N. Mead, E. Hough and T. Stehney, "Security quality requirements engineering (SQUARE) methodology, Carnegie Mellon Software Engineering Institute, Technical report CMU/SEI-2005-TR-009

[13] Moscaritolo, Angela. Most organizations falling short on cloud security policies: http://www.scmagazineus.com/most-organizations-falling-short-on-cloud-security-policies/article/167415/, April 2010.

[14] Y. Yang, Z. Fu, and S. F. Wu, “BANDS: An Inter-Domain Internet Security Policy Management System for IPSEC/VPN”, Journal of Communication and Networks.

[15] H. Abie and A. Skomedal, “A Conceptual Formal Framework for Developing and Maintaining Security-Critical Systems”, IJCNS International Journal of Computer Science and Network Security, Vol. 5 No.12 December 30, 2005

VITA

Kenneth Kofi Fletcher was born on September 7, 1984 in Tema, Ghana. He received his Bachelor of Science degree in Computer Science from the Kwame Nkrumah University of Science and Technology, Kumasi, Ghana in 2006. He was a graduate student in the Computer Science Department at Missouri University of Science and Technology beginning in August 2009. He worked as a Graduate Research Assistant under Dr. Xiaoqing (Frank) Liu from September 2010 to January 2011. He received his Masters in Computer Science and a Certificate in Software Engineering at Missouri University of Science and Technology (formerly University of Missouri – Rolla) in December 2010.