System Quality Requirements Engineering (SQUARE) Methodology: Case Study on Asset Management System

Peter Chen
Marjon Dean
Don Ojoko-Adams
Hassan Osman
Lilian Lopez
Nick Xie

Nancy R. Mead, Advisor

December 2004

System Quality Requirements Engineering (SQUARE) Methodology:
Case Study on Asset Management System

Peter Chen
Marjon Dean
Don Ojoko-Adams
Hassan Osman
Lilian Lopez
Nick Xie

Nancy R. Mead, Advisor

December 2004

Networked Systems Survivability Program

This report was prepared for the

The ideas and findings in this report should not be construed as an official DoD position. It is published in the interest of scientific and technical information exchange.

FOR THE COMMANDER

Christos Scondras
Chief of Programs, XPK

Table of Contents

List of Figures

List of Tables

About This Report

The following report reflects the work of six graduate students, along with Professor Nancy Mead, on their project requirement during the summer of 2004. Given that the System Quality Requirements Engineering (SQUARE) methodology was still in its infancy at the time of this writing, this project served as the first full-fledged and documented implementation on a real-world application.

Even though the SQUARE methodology could be applied by the same team responsible for software development, this report gives feedback on the process from one of the following real-world parties that might be delegated the task of producing the security requirements on a similar application or system:

1. An outsourced consulting or development team.
2. An in-house team or department that is separate from the principal software development team or department within the same company.

The strategy followed in this project simulates the interaction between any of the aforementioned parties with a software development team. The report documents the methodology followed on each step, the approximate amount of time spent, the client's feedback on the process, and recommendations on the overall approach. A few models for decision making under uncertainty are also researched and presented.

It should be kept in mind that the conclusions presented in this report are bounded by the limits of the application and, inherently, the company under study. What applies here might not be relevant on a different system. For example, the company we worked with had only a couple of main software developers and minimal documentation on the application. Consequently, a large portion of time was initially spent in trying to understand the system architecture and interactions, which could have possibly been completely avoided with another client.

Moreover, considerable time was spent on learning about the processes and documentation. None of the team members had any previous experience with requirements engineering, and hence a steep learning curve was followed. For example, in documenting the misuse cases, seven versions were revised before settling on the final one. This means that for the reader, the approximate time allotted per step(s) and efforts invested should have those learning curve factors accounted for. On the plus side, the output of the work done should be viewed as a facilitator to the future group of people who will be working on the SQUARE methodology, giving them more time to spend on researching several other areas.

Even though the process followed could have focused more on researching certain areas, the promises with the client and the limited time available (12 weeks) pushed the team to focus on a tangible deliverable as a primary concern and research as a secondary one. In one example, a few attack trees were drafted after the misuse cases were close to their final stages, but they had to be abandoned due to time limitations, and focus was directed more on models for architectural recommendations.

This report was also complemented by a separate client report (*System Quality Requirements Engineering (SQUARE) Methodology—Application on the Asset Management System*), which included the findings and final recommendations. The material covered in that client report is included in the appendices of this report. Given the large amount of data that was documented throughout the whole process, it was infeasible to include everything in this report. Therefore, snapshots of interim documents were included as figures to exemplify the process.

Acknowledgements

The research project could not been completed without the guidance of Professor Nancy Mead, who not only served as the project faculty advisor, but also encouraged and challenged the team through the duration of the project. The SQUARE team would also like to thank Professor Jonathan Caulkins for providing direction on the mathematical and prioritization models used to analyze the misuse cases. Finally, the SQUARE team would like to thank the Acme Company for providing a software suite for evaluation, research, and implementation of the SQUARE methodology.

Abstract

This report exemplifies the application of the System Quality Requirements Engineering (SQUARE) methodology developed by the Software Engineering Institute's Networked Systems Survivability Program on an asset management application. An overview of the SQUARE process and the vendor is presented, followed by a description of the application under study. The nine-step process of requirements engineering is then explained, and feedback on its implementation is provided. The report concludes with a synopsis of the findings and recommendations for future work.

This report is one of a series of reports resulting from research conducted by the SQUARE Team as part of an independent research and development project of the Software Engineering Institute.

1 Introduction

1.1 Overview

The following section gives some background information on the System Quality Requirements Engineering (SQUARE) methodology, a description of the client for whom the methodology was applied (the Acme Company), and an explanation of the Asset Management System and its applications.

1.1.1 SQUARE Methodology

The SQUARE methodology is a nine-step process developed by Professor Nancy Mead as a part of a research project with Professors Donald Firesmith and Carol Woody to ensure the safety and survivability of IT systems and applications. Although the SQUARE methodology is still under review by the Software Engineering Institute's Networked Systems Survivability (NSS) Program, it demonstrates great potential for industry-wide adoption for developing secure applications and systems. The methodology was applied on the Acme Company's Asset Management System for evaluation, where it assisted in identifying potential threats and vulnerabilities. It also recommended necessary improvements to ensure normal application and system operation in the event of any security breach.

1.1.2 The Acme Company

The Acme Company is a private company headquartered in Pittsburgh with a staff of approximately 1,000 across multiple offices in the United States. It provides technical and management services to various public sectors and a number of diversified private companies.

1.1.3 Asset Management System (AMS)

ABC Services is one of four major subsidiaries of the Acme Company. ABC provides a range of specialized services for asset management. With over 15 years of experience, ABC developed the Asset Management System (AMS). This software product provides a tool for companies to make strategic allocations and planning of their critical IT assets. AMS is an Executive Asset Management Information System that provides decision support capabilities via customized views. These views are displayed in graphical forms and consist of information such as asset information, operational performance, and other user-defined metrics.

AMS also integrates with many third-party software suites to provide enterprise-level services and features. ARCHIBUS/FM, which is used internally, is a facility infrastructure management and operation tool that supports all aspects of infrastructure management. It is also fully integrated with AutoCAD, an industry standard software application that ensures proper change management. All changes made on architectural drawings are immediately reflected in the database. Another integrated tool is a backend Geographical Information System (GIS) used to organize information and geographic locations by sites.

Overall, the AMS Software Suite is a full-service support product in all aspects of infrastructure management and facility-related services.

1.2   Methodology

When the team started working on the project, two final deliverables were kept in mind: a client document (*System Quality Requirements Engineering (SQUARE) Methodology— Application on the Asset Management System*) and a process document (this report). The purpose of the client document was to outline the findings and output of the methodology, whereas the process document was to include client feedback, difficulties encountered, and recommendations on the method used.

With both these deliverables in mind, the team basically worked in parallel throughout the whole project, documenting the process, keeping track of the approximate time spent on each step, researching several methods, and providing the client with interim deliverables. The following gives an overview of the team, the logistics followed, and the framework applied.

1.2.1 Team

It was thought to be a good idea to include some background information about the team members who worked on this project. This would give the reader a better idea about what output to expect given a certain amount of effort invested.

Initially, the team started out as three students in pursuit of satisfying their graduate project requirement under the Master of Science in Information Security Policy and Management program. It then grew to five people, after two other students—one from the Masters of Information Systems Management program and another from the Master of Science in Information Technology program—expressed their interest in joining the project. During the first week of the semester, another two students who were interning at the CERT Coordination Center® (CERT/CC) for the summer also joined the venture, one of whom was directly involved with the team deliverables, and another who was working separately on an online tool to automate the process.

®   CERT and CERT Coordination Center are registered in the U.S. Patent and Trademark Office by Carnegie Mellon University.

In all, seven graduate students, in addition to Professor Nancy Mead, were involved in working on SQUARE and the Asset Management System. Each member of the team had a varied skill set of academic coursework and experiences; some came from very technical backgrounds, where others had their strengths on the policy and management side. However, nearly all had touched upon some coursework in the information security field at Carnegie Mellon University.

1.2.2 Logistics

Given that the entire project was to be completed in a relatively short period of time (three months), a timeline had to be allotted for each step. During the first couple of weeks, the team successfully adhered to pre-set deadlines, but due to the nature of the project, it was quite difficult to follow through with that. As a result, the team decided to focus on completing each step without any set deadlines—even though there was the possibility that not all of the steps could be completed. Fortunately, by strategically assigning tasks to different team members and working on separate steps in parallel, the entire process was implemented.

Throughout the first two months of the semester, the team met twice a week with the project advisor (Professor Nancy Mead), and internal meetings were scheduled on an as-needed basis (usually on weekends). The frequent meetings with the advisor were particularly crucial during the preliminary phase of the project, when guidance was needed the most. However, for the rest of the project timeline, the team met only once per week with the project advisor for high-level follow up and twice internally.

1.2.3 Framework

Initially, the nine-step process was followed on a step-by-step basis using guidelines set forth by the original documentation supplied (Figure 1 shows a summary of the headings for each of the nine steps). During the project, however, some of the steps were lumped together and further subdivided into several categories. This was based on the logical perception of how the process methodology flowed. Even though some of the subcategories were not chronological in the sense that one was a prerequisite of the other, they are presented in this report as such for simplicity. Figure 2 shows the revised steps of the SQUARE methodology and the categories that are represented.

Step 1 – Definitions
Step 2 – Safety & Secuirty Goals
Step 3 – Elicitation Techniques
Step 4 – Artifacts
Step 5 – Initial Requirements
Step 6 – Categorize Requirements
Step 7 – Risk Assessment
Step 8 – Prioritize Requirements
Step 9 – Requirements Inspection

Figure 1:   Original SQUARE Steps

- Definitions   } Step 1
- Safety & Security Goals   } Step 2
- System Architecture
- Use Cases   Steps 3-6:
- Misuse Cases   Artifacts and Initial
- Attack Trees   Requirements
- Prioritization
- Categorizing and Detailing   Steps 7-9:
  Recommendations   Architectural and
- Budgeting and Analysis   Policy Requirements

Figure 2:   Modified SQUARE Steps/Categories

This report is organized based on each of the categories outlined in Figure 2. Every category's purpose is explained, its methodology is described, and the client's feedback on it is outlined. Recommendations and lessons learned on each category are also delineated.

2 Step 1: Definitions

Step 1 consisted of producing a set of definitions agreed on by the client and the SQUARE team. An agreed-on set of definitions enabled both the client and the team to gain a common understanding of information security terms and build a foundation for the system to be analyzed. This was an important step because the client and team initially had varying definitions for the same terminology. It further held both parties accountable for the successive steps based on a sign-off on the definitions.

2.1   Methodology

The methodology followed was quite interactive with the stakeholders and the team members. A primary face-to-face meeting was set with the client to gather information and expectations and to set deliverable dates for the step in order to lay a solid foundation for the SQUARE methodology.

Initially, the SQUARE team met for a couple of hours to brainstorm the preliminary list of definitions, which included popular words from the security industry. This was primarily based on the team's academic courses and prior work experience. The SQUARE team was then divided into two subteams composed of three members each. One team was responsible for researching industry-standard definitions and terminology in well-known resources such as the Institute of Electrical and Electronics Engineers (IEEE), CERT, and the Software Engineering Institute (SEI). The other was in charge of researching supplementary sources such as Web sites (Webopedia.com, Definition.com, Whatis.com, Searchsecurity.com, etc.), information security books, white papers, and published articles. During the research process, several other security definitions were brought up and added to the original list. The entire team was updated on the new terms (primarily through email), and subsequent research was conducted. The process took approximately five days to complete. Some terms that were included were *availability*, *firewall*, *fault tolerance*, *integrity*, and *intrusion*.

The output from both subteams resulted in a document that had multiple definitions (approximately seven) for each term, so the entire team met to cut down the list to around one to two definitions each and to vote on one that would be recommended. This took approximately three hours to complete and an additional four hours to write up the documentation. The team then sought input from the client through submission of the compiled definitions with appropriate referencing. The documentation requested the client to indicate which definitions best fit their needs for security implementation. It also requested the client to modify or comment on the definitions and add any that may have been omitted. Figure 3 shows a snapshot of part of the document that was submitted.

Control:

    a.   Procedures, which can reduce, or eliminate, the risk of a threat becoming an incident. [1]
    b.   An action, device, procedure or technique that removes or reduces vulnerability. [2] [recommended]
    c.   _____
         _____
         _____

Corruption:

    a.   A threat action that undesirably alters system operation by adversely modifying system functions or data. [1] [recommended]
    b.   _____
         _____
         _____

Cracker:

    a.   Someone who breaks into someone else's computer system, often on a network; bypasses passwords or licenses in computer programs; or in other ways intentionally breaches computer security. [13] [recommended]
    b.   _____
         _____
         _____

Denial-of-Service (DoS) Attack:

    a.   One in which a multitude of compromised systems attack a single target, thereby causing denial of service for users of the targeted system. The flood of incoming messages to the target system essentially forces it to shut down, thereby denying service to the system to legitimate users. [14]
    b.   A Dos Attack is a form of attacking another computer or company by sending millions of more requests every second causing the network to slow down, cause errors or shut down. [15] [recommended]
    c.   _____
         _____
         _____

Figure 3: Snapshot of "Definitions Document" Submitted to Client

Later on during the project (Steps 3-6), several other security terms (five in all) were used in the misuse cases that were not documented, and the entire process was repeated for those definitions. This resulted in a comprehensive Definitions Document (Appendix A).

2.2　Client Feedback

The client took approximately one week to decide on a set of definitions. Initially, confusion resulted in a document that was returned with full approval but without feedback or indication of which definitions were agreed on. As a result, the document was then resubmitted and detailed feedback was requested.

The second document returned included feedback on the following (red comments in Figure 4):

- selection of definitions (in checkmarks)

- comments on definitions that seemed ambiguous (such as that for "Downtime")

- modification of certain definition terms

- additional terms that were not originally included

Downtime:

a. The amount of time a system is down in a given period. This will include crashes and system problems as well as scheduled maintenance work. [18] [recommended] ✓
Don't know if I agree with including scheduled work during non-production hours in the downtime calculation. It's a "If a tree falls in the forest and no one is around to hear it, does it make a sound" scenario?

Espionage:

a. The act or practice of spying or of using spies to obtain secret information, as about another government or a business competitor. [20] [recommended] ✓
b. _____
_____
_____

Port Scanning:

a. A series of messages sent by someone attempting to break into a computer to learn which computer network services, each associated with a "well-known" port number, the computer provides. [1]
b. The act of systematically scanning a computer's ports. [34] [recommended]
c. I agree with both of the above definitions
_____
_____

Access Control List:

a. A table that tells a computer operating system which access rights or explicit denials each user has to a particular system object such as a file directory or individual file. [2] [recommended] ✓
b. _____
_____

Please indicate any definitions that have been left out from the list:

a. Spoof – Same as masquerade?

b. _____
_____

Figure 4:   Snapshot of "Definition Document" Comments from Client

There were also definitions that seemed to overlap and have double meanings, for which clarification was necessary through phone and email. The client relied on the SQUARE team's expertise and knowledge for guidance through the first step. As a result, most of the team's recommendations on the definitions were selected.

The same feedback was given for the five definitions that were submitted later on during the project (Steps 3-6). This resulted in a final list of definitions that was agreed on by the stakeholders and submitted to the client.

2.3   Recommendation

Step 1 is considered to be a necessary prerequisite to the successive SQUARE steps. It enables all stakeholders to bridge a potential miscommunication gap. Each party, based on background and experience, could have varying knowledge and perceptions of security terminology. It is therefore important to clarify terms before providing further consultation and recommendations.

The methodology followed seemed to be quite effective and would be recommended for any future applications of the SQUARE methodology. In the case of the Acme Company, the SQUARE team only dealt with a single person from the technical side who confirmed all the definitions, which made it a fairly easy step both in terms of time and effort. In reality however, this might not apply in cases where there is more than one stakeholder involved. In more complex systems, for example, the entire internal technical staff would need to deal with security nomenclature and should therefore be involved in agreeing on the definitions. Even with the Asset Management System specifically, a more refined output would be one that included the end users of the system—probably the insurance companies and client organizations within the industry.

Another point worth mentioning is the need to establish who the stakeholders are as a pre-phase to Step 1. During the first meeting with the client, it was important to define who the individuals responsible for the decisions were. This allowed for effective communication and change commitment. The method was also efficient because there was only one point of contact from each side (the SQUARE team and the client); this would be highly recommended in the case where documents and changes need to be approved by several parties.

Regarding the first submitted document, the decision to cut down the definitions from approximately seven per term to one or two was based on the idea of minimizing work on the client side. However, it might make more sense to have more than that, depending on the complexity of the system under study.

Given that the process was repeated later on in the project (Steps 3-6), it was shown that the learning curve had a substantial effect, especially in terms of research and documentation. Having completed the process before, the team did the research part in less than a day (mainly because the sources were already there) and the documentation in around an hour.

Future teams could make use of the definitions in Appendix A as a template for the required definitions. However, it is advisable to verify the references, given that URLs might have changed. Overall, Step 1 consumed relatively little time and was executed with few to no flaws.

3 Step 2: Safety and Security Goals

Step 2 consisted of identifying the client's security needs regarding the Asset Management System. This included objectives that the client desired to accomplish as a result of the deployment of the SQUARE process. A series of measurements to ensure the safety and survivability of the client's system were implemented. Also, the system's stakeholders, core components, and services were identified. This will assist Acme Corporation in implementing the necessary security measures to ensure the survivability of the AMS software suite. Security services affected by security incidents, such as confidentiality, integrity, and availability, were identified and categorized by the access level rights of the user. Using this process assisted the SQUARE team in analyzing system security requirements needed to ensure the system's overall security and survivability. By analyzing Acme's safety and security goals for the Asset Management System, the SQUARE team was able to establish a security foundation in order to justify its discoveries and recommendations in the successive steps of the SQUARE methodology.

3.1   Methodology

The client originally supplied a couple of papers outlining the Asset Management System's business objectives and functional requirements. Very little information was available on system interactions or architecture, given the relatively small size and seemingly minimal resources of the company. In response to that, the SQUARE team submitted a paper to the client that outlined what the major security goals should incorporate.

The points presented were generic rather than specific to the Asset Management System. The first document had an overview of the mission objectives and a stakeholder analysis. The major security objectives (confidentiality, availability, integrity, etc.) were also included, along with a set of questions regarding what needed to be addressed. For example, data integrity as a security objective was presented as shown in Figure 5.

Data Integrity

Integrity of data is absolutely critical in the Asset Management System package. If the underlying information upon which facility managers must make their decisions is corrupted or wrong, the purpose of the package has been defeated. That solidifies that the issue of data backups and checksum integrity verification are of the utmost importance.

Some questions that need to be answered are: What integrity verification controls are in place? How long are backups maintained to verify integrity? What change management processes are in place to ensure that alterations are approved and reviewed by the proper decision makers?

Figure 5:   Example Showing How "Data Integrity" Was Presented

Foreseeing some difficulty with outlining the stakeholders and their responsibilities, the team felt there was a need to make sure that the system architecture and network topology were well understood. The team created preliminary diagrams describing the system interactions and then confirmed them with the client to ensure that all the connections were logically correct. Figure 6 and Figure 7 show the network topology and system architecture diagrams approved by the client.

TYPICAL NETWORK TOPOLOGY FOR ASSET MANAGEMENT SYSTEM
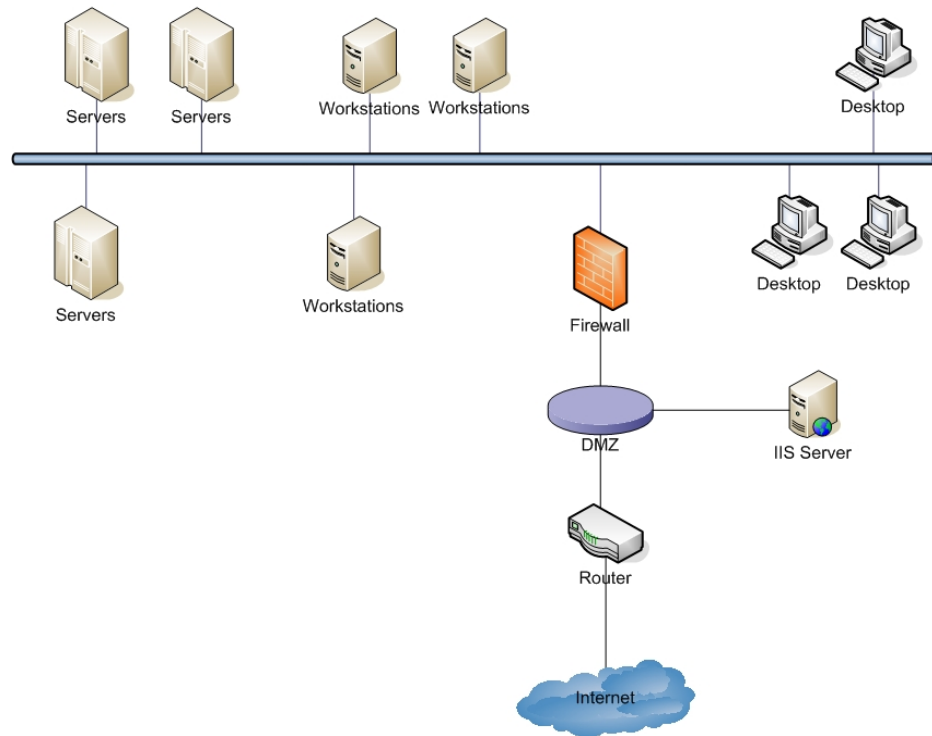
Figure 6:   Network Topology Diagram of Asset Management System
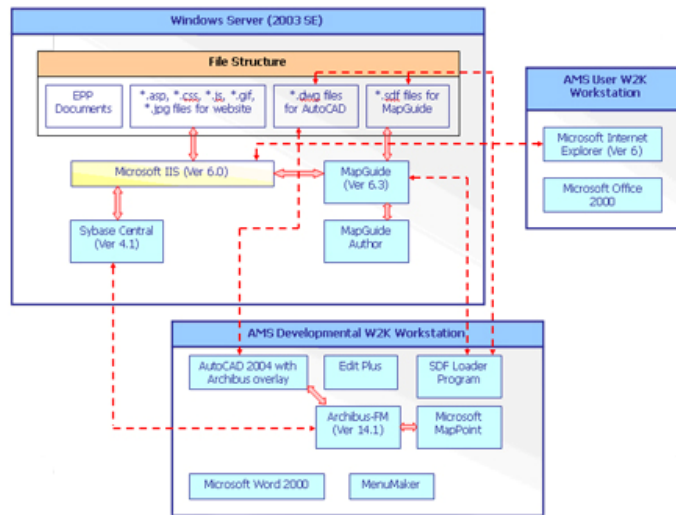
Figure 7:   Preliminary System Architecture Diagram of Asset Management System

Depending on the end user, the Asset Management System could have multiple system and network architectures. However, for simplification, the team assumed the structure of the Asset Management System as it was implemented on the client, and analyzed it accordingly.

Moreover, the SQUARE team decided not to address any migration plans that the client had in mind, based on the fact that doing so would complicate the analysis process.

The first draft of the architecture diagram was intentionally sketched out without any legend or explanation of the connections. This was mainly done to speed up the process and get feedback on the system from a high-level perspective, and resulted in the team outlining the safety and security goals. Moreover, given that detailed architecture diagrams usually take some time before they are finalized, it was thought to be an efficient strategy to start with a rough draft, get the client's approval, finish up work with Step 2, and keep the process of updating and modifying the system architecture running.

Based on the network and system diagrams, the stakeholder analysis section was further revised and stakeholders were divided into four main user levels: High, Medium, Low, and System Administrator (Authority/Access). In addition, a couple of meetings with the client helped modify some of the security objectives to adapt for system details. Figure 8 shows a snapshot of the document that defines the High-Level System User:

High-Level System User

These users will have read, write, modify, and delete access permissions in the AMS developmental workstations. To be granted high-level access, the employee/personnel must be involved with the maintenance and support of all the modules/components within the developmental workstations, but not to the Windows Server. Modules/components include

- Archibus Facility Management. This includes event logs, database entries, and storage.
- Time and Attendance System. Review the inputs of data in the system.
- Facility Drawings and Procedures. Ability to make updates and modifications of facility drawings, policies, and procedures.
- Event Logs. Users will be able to perform maintenance and review of all event logs stored in the Sybase databases.

Figure 8:   Snapshot of Document Describing High-Level System User

From the diagrams and stakeholder break-down analysis, preliminary misuse cases and their corresponding schematics were produced to define specific scenarios that would be counteracted by security measures. Figure 9 shows an example of a misuse case schematic, in which a malicious user deletes critical system data.
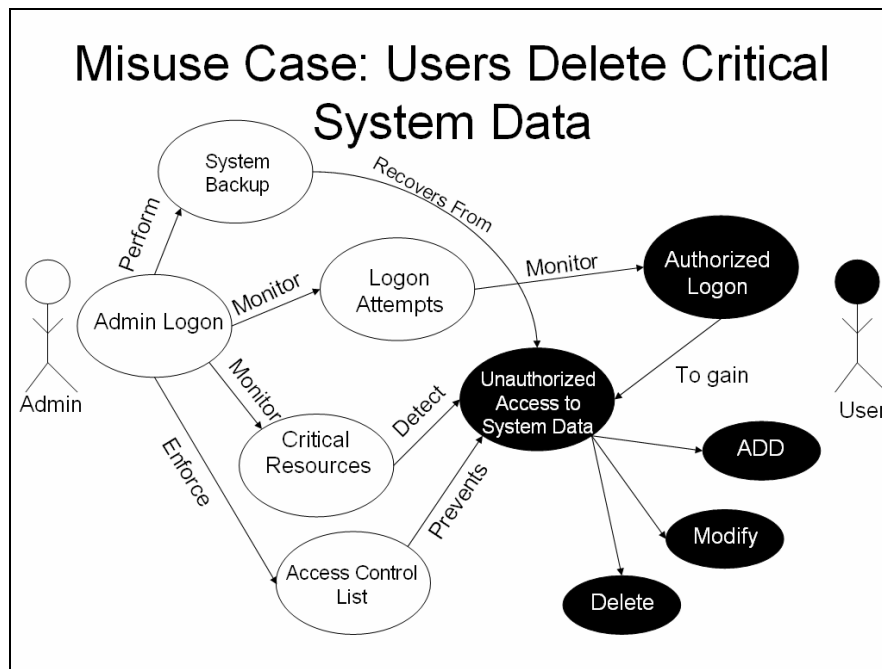
Figure 9:   Misuse Case Schematic Example

Even though the misuse case schematics were not considered a direct requisite of Step 2, they were quite helpful in understanding the overall interactions of the system. They also helped the team anticipate requirements of later steps in the methodology.

The client was presented with a final document that contained the high-level security and safety goals, a description of the stakeholders, and the network and system architecture diagrams (the misuse case schematics were not included in the document). Overall, the process took approximately two weeks to finish, which included a face-to-face meeting and several emails and phone meetings with the client.

3.2   Client Feedback

3.2.1   System Architecture

The team's primary concern in Step 2 was accurately defining the stakeholders, and as mentioned earlier, only a rough draft of the system architecture diagram was drawn. However, there was obviously a need to produce the actual system architecture diagram complete with call/callee interactions, access control types (read/write) and connection types (file access/data access/control transfer). Several phone meetings with the client were needed to ensure that all the connections were logically correct and that all the modules and components were in the right place. It took around five weeks from when the initial draft of the system architecture was drawn until approval of the final one. This was mainly due to the fact that the lead technical person responsible for approval was unavailable most of the time. Appendix D shows the final draft of the system architecture diagram and its legend.

3.2.2 Stakeholders

Initially, the stakeholders were divided into three main levels of users: High, Medium, and Low. Each corresponded to a module in the system architecture diagram. However, the client asked for a revision of the stakeholder categories, and as mentioned earlier, four rather than three levels were used. A System Administrator was added, which corresponded to the person responsible for the Windows 2003 Server and all corresponding installations, updates, and modifications of Microsoft IIS, the Sybase DBMS and file structures. For a complete description of all the stakeholders and their access right levels, please refer to Appendix B.

3.2.3 Objectives

The security objectives questions that were presented in the first document were answered by the client. There were some misconceptions regarding what was needed, and most concerns were answered through phone and email. The client replied back with the final answers to the security objectives after around a week. Figure 10 shows a sample answer (in red) given by the client.

<u>Monitoring</u>

One security goal is to preserve or enhance the ability to accurately record the activities that take place. When users interact with the system, we would like to have a complete accounting of all commands issued, as well as the internal transactions of the package. In order for this to happen we need to know the Logging Capabilities that are currently in place for the Asset Management System.

Some questions that need to be answered are: What are the logging capabilities of Windows Server, IIS 5.0, ASP, Autodesk MapGuide, and other required components of the Asset Management System?

Initially, full logging should be maintained for all AMS applications until installation and acceptable performance goals have been reached. If disk space becomes an issue, the log file content and/or retention time may be reduced as long as security breaches are still captured and application messages are at such a level that they can be used to debug errors quickly.

Figure 10: Sample of Security Objective Reply Given by Client

3.3   Recommendations

In general, it was quite a challenge to outline the security objectives and stakeholders of the system before understanding the architecture. This raised a few important questions for future applications of the SQUARE methodology: Should there be an existing system set in place before the high-level goals are defined? And if so, how far through the software development

---

process would be the ideal point to analyze the security of the system? These questions would be worth researching in future applications of the SQUARE methodology.

In this particular case however, it was recommended to have a substep between Steps 1 and 2 that required defining the system architecture. As mentioned before, the client was an exception in the sense that no previous documentation was available on the system interactions, and drawing the system architecture diagram took much more time than expected.

Based on that, the team moved on to working on the subsequent steps while the system architecture diagram finalization was still underway. Another lesson learned about the system architecture diagram, however, was that the call/callee interactions between the system components were added in only during the final stages and cleared up the system interactions considerably. It probably would have been more effective if the team had started working on the call/callee interactions at an earlier stage.

There were some difficulties encountered with outlining the security goals. In the initial document, the goals outlined by the client were so high level that they were close to being definitions rather than objectives. Although the security goals should be from a high-level perspective, they should still apply to the system under study. It is important to note that the way in which the questions were presented to the client defined the granularity of feedback required. It was much easier for the client to answer specific questions rather than general ones. Moreover, giving some sample answers also helped tremendously in pinpointing the security objectives of the system and should be taken into consideration for future applications.

Misuse case schematics were used in this step to identify possible intrusion scenarios for the AMS software suite. Through identification, this allows the team to compile comprehensive misuse cases/scenarios in the latter part of the SQUARE process. Given that misuse cases (see Section 4.2, "Misuse Cases") were drawn later on in a different way, this version of the misuse case schematics provided a bigger picture of possible intrusions that may affect the survivability of the AMS software suite.

4 Steps 3-6: Artifacts and Initial Requirements

4.1 Use Cases

Step 3 in the SQUARE process entailed drafting use cases (nine in all) and their respective diagrams with the assistance of program managers at the Acme Company. Use cases describe a list of interactions between the user and the system under review to achieve a goal. A use case comprises

1. the user who interacts with the system, described as an actor
2. a description of the goal to be achieved through the use case
3. assumptions that must be met for the use case to be completed successfully
4. a listing of the actual steps between the actor and the system
5. variations or alternative paths to achieve the goal
6. non-functional requirements that the use case must meet, such as performance or reliability

Use cases specify a range of ways to use the system. They define the functionalities required of the system. They were helpful in identifying mission-critical services and their underlying assets.

4.1.1 Methodology

The methodology implemented for the use cases was an interactive approach involving the SQUARE team and Acme that stretched over about a one-month period, with two client site meetings and conference calls. The team was able to analyze the system through a Web-accessible client deployed by Acme for the SQUARE team to interact with. This included a URL address and a logon window with username/password authentication to enter the system. An initial draft of the use case profiles was generated based on data gathered on the common functions performed by the Asset Management System and how users navigated through the system. When accessing the system, the team discovered that it did not contain sufficient navigation capabilities to generate an accurate set of use cases or related diagrams. Therefore, the process halted when the SQUARE team came upon some user-system interactions that could not be accounted for. As a result, members arranged for an on-site client meeting to engage in a face-to-face meeting with the system developers. Members of the SQUARE team accessed the Asset Management System to determine the primary and

alternate routes through the system and data on the common functions it performed. The lead developer explained the various functions performed by the applications contained in the system and their contribution to the overall successful completion of its mission. Members of the SQUARE team were then permitted to navigate through the system (with the assistance of the developer) to determine the primary and alternate routes through the application to achieve a specified use of the system. During the system walkthrough, one team member was responsible for navigating the system while another kept a record of the actions taken to achieve the desired result. The data gathered from the session was then entered into the use case template (Table 1) based on the categorization of attributes. The team then distributed the early drafts of diagrams and use case profiles to verify that the use cases correctly reflected the interaction of external users with the system. The process then became iterative in that the use case profiles and diagrams were proposed, reviewed by the SQUARE team, edited based on the suggestions of the team, and deemed ready for final presentation to the Acme Company or in need of further editing.

After the generation of the misuse cases, the authors of the misuse and use cases then conferred to determine the appropriate links between the relevant misuse cases. The use cases were then presented for feedback and final validation. Table 1 gives a description of the attributes in the use cases.

Table 1:   Use Case Template

Number	*Use case identifier and reference number and modification history*
Use Case	*Use case title*
Description	*Goal to be achieved by use case and sources for requirement*
Actors	*List of actors involved in use case*
Assumptions	*Conditions that must be true for use case to terminate successfully*
Steps	*Interactions between actors and system that are necessary to achieve goal*
Variations	*Any variations in the steps of a use case*
Non-Functional	*List of any non-functional requirements that the use case must meet*
Related Misuse Cases	*List of any related misuse cases that may be affected by this use case*

Use case diagrams (visual representations of the user's interactions) were generated based on the goals described in the use cases. These diagrams depicted the steps needed to successfully achieve the stated goal. The diagrams also identified which Asset Management System assets were accessed in the use case. Figure 11 shows a use case diagram example.

UC-03: Mark Up/ Create Floor Plans
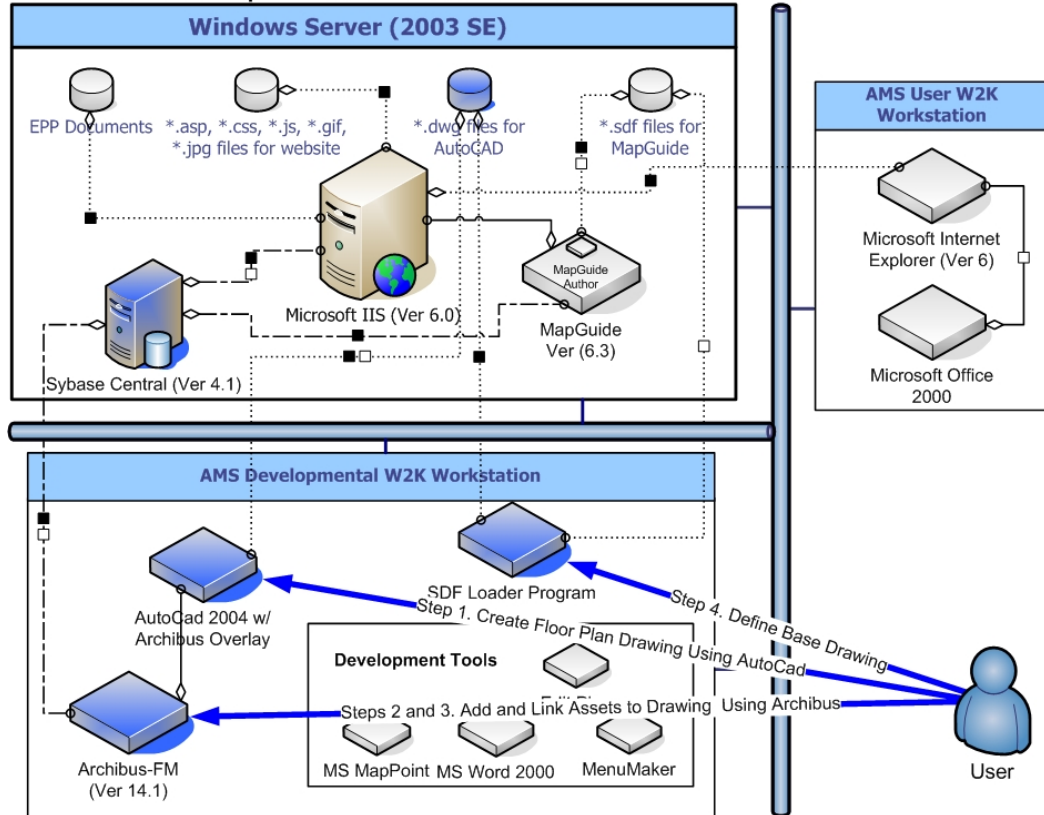
Figure 11: Use Case Diagram Example

4.1.2   Client Feedback

When detailed accounts of functionality steps were not clear enough to produce a defined use case, the client was asked to review and provide suggestions and guidance for counteracting those uncertainties. For example, the SQUARE team was given limited, supervised access to the Asset Management System. As a result, the team was not able to provide detailed steps for installing the Asset Management System. The client provided a detailed account based on a general AMS installation process (Table 2) that does not reflect on any specialized system configuration for the AMS client.

Table 2:   Sample Use Case: UC- 06 Install the Asset Management System

Number	UC- 06
Use Case	Install the Asset Management System
Description	System Administrator wants to install the Asset Management System on the network
Actors	System Administrator

Assumptions	The Sys Admin has control over the network

Steps	Steps for Pre-Determined Windows Server(s)
	
	 1. Install/confirm IIS.
	 2. Install/confirm Map Guide with Map Guide Author option.
	 3. Install/confirm database engine (Sybase or access control list, etc.).
	 4. Copy client database file to server (assuming that client database file was previously created and configured).
	 5. Configure ODBC System DSN and confirm connectivity to database.
	 6. Confirm that line in vbdefs.asp references the configured ODBC System DSN name.
	 7. Configure Web site in IIS:
	   - Assign Web site name (e.g., Asset Management System).
	   - Associate with IP address assigned to server.
	   - Do not allow anonymous access.
	   - Specify Integrated Windows authentication.
	   - Specify home directory path.
	   - Specify default content page.

The client also provided a detailed account for creating links to the system, which the SQUARE team was unable to perform while on the client site (Table 3).

Table 3:   Sample Use Case: UC- 07 Create Links

Number	UC-07
Use Case	Create Links
Description	High-level users will have the ability to access the Asset Management System and create links to EP procedures, docs, etc.
Actors	High-Level User or System Administrator
Assumptions	This assumes that • System Admin has added write privileges to the access control list (ACL) of the document repository folder • system is available • data entered is correct
Steps	1. User logs into developmental workstation with assigned network username and password. 2. The system authorizes and authenticates the user and the user is allowed into the system. 3. User enters data into ARCHIBUS/FM tables "ep_procedures" and "ep_bl_doc_link" to denote document path, document name, etc.

After all of the use cases and their corresponding diagrams were compiled and completed, they were sent to the client for approval. The client then provided needed alterations and

editing. This final editing allowed for the generation of an applicable and practical set of use cases.

In an otherwise non-developed system, generation of use cases might not be possible. However, since the Asset Management System and its architecture were available for analyzing, the SQUARE team was able to construct a set of applicable use cases and diagrams. For the complete set of use cases and their diagrams, please refer to Appendices C and D respectively.

4.1.3 Recommendation

The SQUARE team would recommend differentiation of the SQUARE use case step between a developed system and a non-developed system. There should be an additional step in the process to analyze use cases. If the client has a system architecture, the SQUARE process should incorporate a step that includes creation of a set of use cases. On the other hand, if the client does not have an intact system architecture, the usefulness of the use cases would be limited, and use cases may be unrealistic to implement. In such a case, the SQUARE process may not need a use case step.

4.2 Misuse Cases

Step 4 of the SQUARE process consisted of generating misuse cases that could occur to the Asset Management System. Misuse cases and diagrams were identified and agreed upon by the client and the SQUARE team. The purpose of identifying potential misuses of the application was to recognize any vulnerabilities in the existing Asset Management System architecture and provide a set of architectural and policy recommendations to mitigate those vulnerabilities. In addition, misuse cases also contributed to the prioritization of system functionalities in the Asset Management System, particularly those integral to the minimally required system functionalities in the event of security breaches (i.e., survivability). Misuse cases assisted Acme in identifying possible threats and provided architectural and policy recommendations to secure its critical Asset Management System components.

4.2.1 Methodology

Step 4 was initiated with a preliminary meeting with the client to understand the existing system architecture of the Asset Management System. Next, after reviewing the agreed-on security definitions (Step 1), security and safety goals (Step 2), system architecture (Step 2), typical network topology (Step 2), and use cases (Step 3), the SQUARE team proceeded to research possible misuses, attacks, and threats that could affect the Asset Management System.

Initially, the team collaborated for two meetings to brainstorm possible misuses and attacks. Web sources (CERT/CC, Microsoft, etc.) were utilized to perform in-depth vulnerability assessments on Web applications and systems such as the Asset Management System. In

addition to vulnerability assessments, the SQUARE team researched other available misuse case templates on the Web. The team discovered that current research on misuse cases could not be leveraged because it did not depict detailed and multifaceted aspects of misuse cases. Therefore, the team decided to incorporate ideas from multiple research sources and to create a comprehensive misuse case template. After a week of research, the team created an initial misuse case template (version 1.0) and used it to create 18 misuse cases. Table 4 shows an example of a misuse case using version 1.0 of the misuse case template.

Table 4:   Misuse Case Template 1.0

Name:	Unauthorized logon on the Windows 2003 server	
Mis-actors:	Unauthorized users	
Security Attributes Affected:	Confidentiality, integrity	
Description:	An unauthorized user attempts to log on to the Windows 2003 server and succeeds.	
Pre-conditions:	1. The unauthorized user has unintended logon rights to the Windows 2003 server. 2. The Windows 2003 server resides on an intranet network.	
Assumptions:	The user does not have expressed permission to log on Windows 2003 server.	
Post-conditions:	Worst Case Threat:	The unauthorized user logs onto the Windows 2003 server machine. His/her actions are never caught.
	Wanted Capture Guarantee:	The unauthorized user never logs on to the machine.
	Wanted Prevention Guarantee:	Enforce machine ACL security policy.
	Wanted Detection Guarantee:	Logon attempts are logged and viewed by system administrators.
Related Business Rules:	1. Any logon attempts should be logged. 2. Unauthorized users should not be able to log on to the Windows 2003 server machine	
Potential Mis-actor Profiles:	Medium to highly skilled; potentially host administrators with medium criminal intent	
Stakeholders and Threats:	AMS client company: loss of data integrity and/or confidentiality if the mis-actors gain access to core services	
Scope	User access concerns	

The subsequent four weeks consisted of several revisions of the misuse case template resulting from feedback from team members and Dr. Mead and from knowledge gained from system functionality documents and AMS demos. The misuse case document was expanded to include four more misuse cases and seventeen categories of information. Table 5 depicts a sample of the finalized misuse case template and categories.

Table 5: Final Misuse Case Template

Misuse Case Categories	Explanation
Number:	*Misuse case number. Each misuse case represents a single threat/vulnerability of the Asset Management System component (Windows 2K server, AMS User Workstation, etc.)*
Name:	*The name of a particular threat/vulnerability. Threat(s) may derive from user or system component interaction.*
Scope:	*System Vulnerability Concern(s)*
Priority:	*Security priority levels for Acme Company. The priority levels address what threats/misuses can affect the Asset Management System processes/operations.* ___ Low ___ Medium ___ High
Deployment Environment:	*Environment/topology of the network on which the affected Asset Management System component is deployed* ___ Intranet *(LAN, etc.)* ___ Extranet/Internet *(Internet, VPN, etc.)*
Mis-actors:	*Attacker type*
Access Right Levels:	*Asset Management System user access rights level. Other network user address users connected to the Asset Management System via VPN.* _x_ Low-Level System Users _x_ Medium-Level System Users _x_ High-Level System Users _x_ Sys Admin _x_ Other Network User
Point of Entry:	*The point of entry at which the attacker gains unauthorized access to the Asset Management System.* ___ Network _x_ Host ___ Application
Security Attributes Affected:	*The security attributes affected by the intrusion/attack.* _x_ Confidentiality _x_ Integrity ___ Availability
Description:	*The description of the attack or system misuse.*

Sophistication:	*The intrusion/attack sophistication levels.*	
	___ *Low: Attacker uses simple-to-use malwares or access to unrestricted/unsecured system resources to perform attacks (hackers, crackers, etc.)*	
	___ *Medium: Attacker uses pre-made attack scripts, malwares, etc. to perform attacks (e.g., script kiddies)*	
	___ *High: Attacker generates own attack scripts, malwares, etc. to perform a series of attacks that may be detrimental to the Asset Management System (professional hackers, etc.)*	
Pre-conditions:	*Initial system and network configuration prior to intrusion/attack*	
Assumptions:	*Existing system and network configuration settings (existing user ACL, system security mechanisms-passwords, etc.)*	
Post-conditions:	Worst Case Threat:	*Threats/attacks occur to the targeted system (user gains unauthorized access to the system undetected, deletion or modification of data, etc.)*
	Wanted Prevention Guarantee:	*Prevention recommendation for the system (i.e., technical recommendations to prevent future attacks)*
	Wanted Detection Guarantee:	*Detection recommendation for the system (i.e., technical recommendations to detect future attacks)*
	Wanted Recovery Guarantee:	*Recovery recommendation for the system (i.e., technical and policy recommendations to recover from attacks)*
Potential Mis-actor Profiles:	*Attacker profile (i.e., attacker characteristics)*	
Stakeholders and Threats:	*Stakeholders affected by the intrusion/misuse. Also includes post-attack threats that may occur for the Asset Management System client company (loss of reputation, clients, etc.)*	
Related Use Cases:	*Use cases affected by the attack/misuse*	
Related Threats:	*Related threats that may be used by the attacker to perform additional misuse/attacks*	
Architectural Recommendation:	*Asset Management System architectural recommendations are to prevent and to detect this particular attack/intrusion (i.e., technical architectural recommendations)*	
Policy Recommendation:	*Recommended policies to assist the Asset Management System client company to better safeguard against this particular attack/intrusion. Policy recommendations include routine updates of systems, configuration files, IT system usage policies, reviews, etc.*	

Misuses were derived from possible unintended (or maliciously intended) usage of the system. A significant amount of time was spent on the formal categorization of misuse impacts, points of entry to the system, misuse sophistication, and user access rights levels. The team consulted various books and reference materials that described common threats and countermeasures to Web-based attacks and misuse examples. These materials provided significant insight into the generalization of misuse cases. They also provided expert-

recommended prevention, recovery, and detection guarantees with which misuses can be mitigated. The guarantees provided the foundations of subsequent architectural recommendations and policy recommendations. Once guarantees were conceptualized, the efforts to translate them into architectural and policy recommendations were uncomplicated. The misuse cases that shared similar threats and recommendations were combined together and generalized into one misuse case. These efforts reduced the scope of the document and minimized repetition of misuse patterns. Below is a sample of the misuse case categories: architectural recommendations and policy recommendations.

Table 6:   Sample of Architectural and Policy Recommendations

Architectural Recommendation:	• (AR-01) All shared drives on the network should enforce authentication policies. • (AR-03) Audit information is stored in a separate location from the servers and the workstations. • (AR-19) Implement role-based authentication control.
Policy Recommendation:	• (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-04) Applications and operating systems must be patched routinely. (Bimonthly) • (PR-07) Enforce strong password policies. • (PR-13) Password-protect any necessary shared documents. • (PR-16) Require users to change their passwords periodically. (Monthly)

The methodology for developing the misuse case diagrams was an iterative process between the Acme Company and the SQUARE team. The initial system architecture was mapped for Acme. Acme provided knowledge about their Asset Management System's system components, interconnections, and communication paths through their network. The SQUARE team used Acme's descriptions to generate a candidate network topology. The candidate topology was then provided to Acme for feedback to ensure overall accuracy.

Once Acme agreed that the proposed system architecture was an accurate representation of their system, the SQUARE team initiated merging the misuse case profiles with the network architecture. An initial draft of misuse case diagrams was generated based on potential vulnerabilities profiled in the misuse cases. Early drafts of diagrams were evaluated by the SQUARE team as to determine whether the diagrams accurately reflected the vulnerabilities profiled in the misuse cases. The process was iterative in that misuse case diagrams were proposed, reviewed by the SQUARE team, edited based on suggestions from within the team, and validated before final presentation to the Acme Company. If further editing was required, the process repeated itself from initiation stages. Misuse case diagrams were presented to the Acme Company along with misuse case profiles for feedback and final approval. Figure 12 and Figure 13 show the final versions of the misuse case diagram legend and diagram.

LEGEND
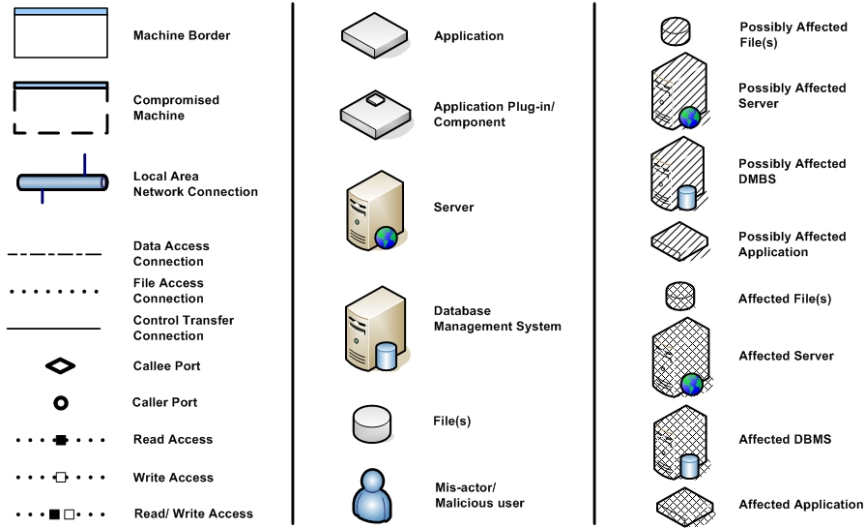
Figure 12: Sample of the Final Version of the Misuse Case Diagram Legend
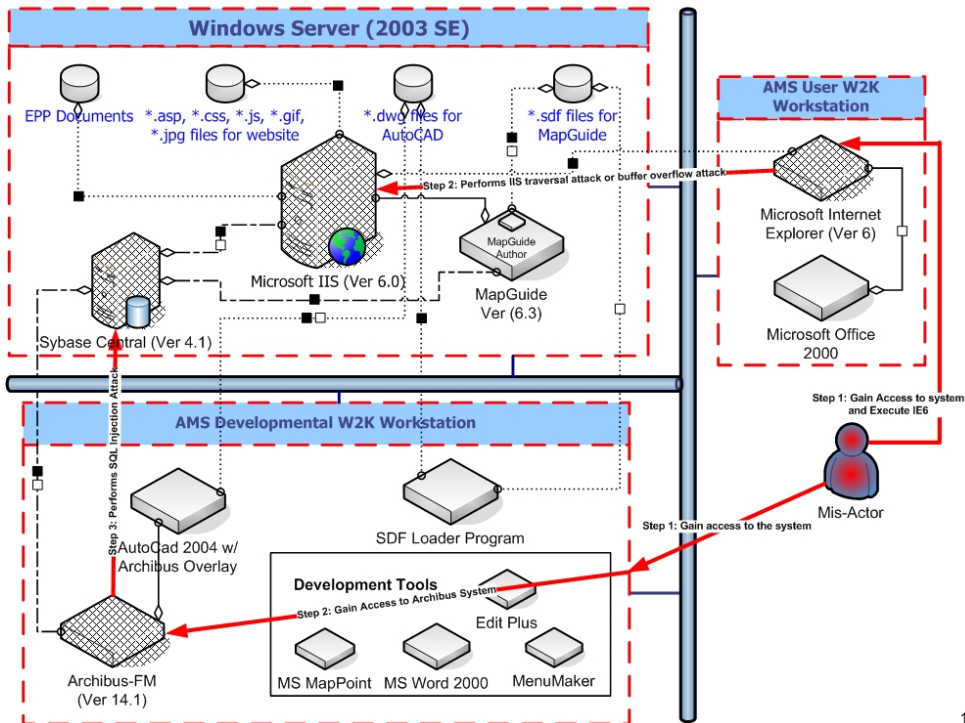
Figure 13: Sample Final Version of a Misuse Case Diagram

4.2.2 Client Feedback

Initially, the SQUARE team created a set of generic descriptions of mis-actors and presented them to Acme. The client suggested that the team differentiate mis-actors based on the access levels that users could be assigned. The client had questions about the differences between Capture Guarantee, Prevention Guarantee, and Detection Guarantee To Acme, there appeared to be many overlapping measurements. This problem was partly due to the team's lack of understanding about the AMS system. As the project progressed and team members gained insight into the system's functionalities, the SQUARE team made enhancements to these areas based on the client feedback. As a result, significant improvements to the system's assessment were made, and the client attained a deeper understanding of their system in uncharted areas.

The client was extensively involved in the verification of misuse cases. Their Web-based system exhibited many of the security problems commonly seen in other Web systems. The client admitted to knowing about many of these common problems but had not correlated them into a formal document. They appreciated the team's effort on the misuse cases because it provided a method to express misuses in written form, which previously had not been available.

The client preferred having the misuse cases in both tabular and graphical format. The diagrams were better visual aids to conceptualize the process to other business users. However, many of the intricacies of the misuse cases could be shown only in the tabular format. The SQUARE team observed that the client had the tendency to immediately scroll to the recommendation section and view the solution to the misuse case. Given that most clients typically have some knowledge of the types of misuse cases affecting their systems, this may be typical: ignoring causal descriptions to attain the immediate gratification of "fixes."

4.2.3 Recommendation

The SQUARE team recommends that misuse cases be created in the early steps of the SQUARE process, after network topologies and use cases are generated. This will enable stakeholders to better understand the threats that may affect their system or application. Thus, the team recommended that the client undergo security threat assessments (misuse cases) in the early stages of the system development life cycle to ensure the survivability of their system. For products still in development stages, it is recommended that stakeholders research common security problems their architecture could be exposed to and generate generic misuse cases from them. For a developed product such as Acme's Asset Management System, common problems are still likely to occur. However, there will also be other product-specific problems that cannot be understood without using the system. Therefore, it is recommended that system demos and use case scenarios be used extensively for products that are already implemented and in use.

The misuse case process typically takes several weeks. It is a labor-intensive process that requires input and buy-in from all stakeholders in order to properly assess the system. Therefore, it is highly recommended that proper expectations be set at the initial meeting. There could also be confusion as to the purpose of misuse cases. It is important to note that misuse case analysis is not limited to finding faults within the system but also includes identifying weak human and process management areas within an organization.

Since Acme has already rolled out its Asset Management System to several clients, the generation of the misuse cases allowed Acme developers and managers to identify potential security threats and vulnerabilities that may affect their system. As a result, Acme will make the necessary modifications through our misuse cases and architecture and policy recommendation to ensure the overall survivability of their Asset Management System.

4.3 Attack Trees

An attack tree is a formal approach to examine a misuse case and to verify that the misuse case's architectural and policy recommendations can sufficiently address all the potential vulnerabilities that can lead to the misuse happening. It is a hierarchical representation of many types of related security breaches on which the misuse case is based. It provides the means to translate from high-level descriptions to detailed case-by-case scenarios of possible security breaches. Each scenario in the attack tree should be examined in detail to see if any set of existing recommendations can sufficiently mitigate its risk. If there is a scenario that is not currently covered, additional architectural and/or policy recommendations would need to be considered and added to the recommendation list. In another words, an attack tree is a detailed visualization of a misuse case and an important element of validation for the architectural and policy recommendations of the misuse case.

4.3.1 Methodology

Attack trees were formulated through a reiterative top-down process in which the SQUARE team visualized scenarios from the perspective of an attacker. The team presented themselves with the question What would it take to cause a misuse case to happen and under what circumstances? The answers provided clues to the discovery and formulation of many levels of scenarios in the attack trees. For each scenario in the level, the SQUARE team examined the actions and the circumstances that would cause that scenario to happen. This process happened recursively until all scenarios were exhausted. At this point, the team verified that existing recommendations sufficiently mitigated the risks. By cross-validating a misuse case's attack tree against its recommendation list, the team gained confidence in the robustness and comprehensiveness of its recommendations. The formulations of attack trees provided justification for the set of to-be-implemented recommendations. Given the time constraints of the project, the SQUARE team produced attack tree diagrams only for a portion of the misuse cases (work on the other misuse cases can be completed at a later date).

Figure 14 shows an example of an attack tree diagram. For the rest of the set, please refer to Appendix G.
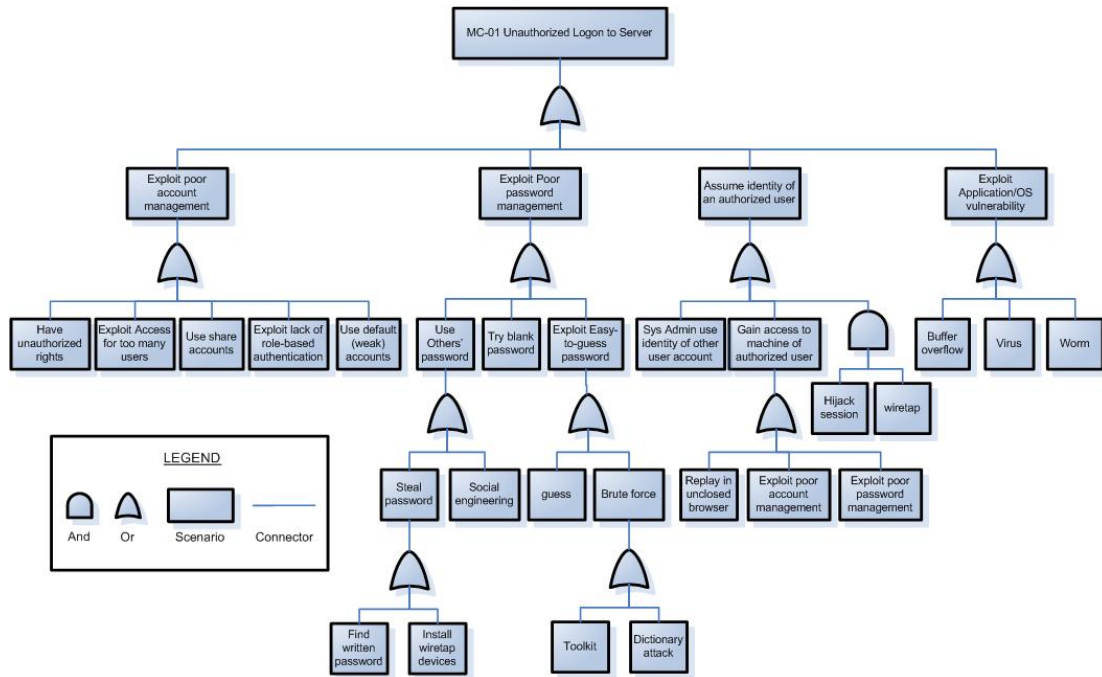
Figure 14: Attack Tree Example

4.3.2 Client Feedback

Client feedback was very limited given the time constraints near the end of the SQUARE project. There were other, higher priority tasks that took the majority of their time. Our experience with the client was that they did not have the necessary security expertise to make extensive use of the attack trees. They were happy to learn that we validated the comprehensiveness of our recommendations against formal representations of the misuse cases, and that was the extent of their involvement in this process.

4.3.3 Recommendation

Attack trees may serve as an effective method to formally represent a misuse case. Their existence in the process is necessary to validate that the misuse case's recommendations are robust against all possible areas of exploits. An attack tree should be drawn as soon as its corresponding misuse case is developed, before architectural and policy recommendations are formulated. After the recommendations are drafted, attack trees can be used to verify the robustness of the misuse case's recommendations. In the final client document, attack trees could be included in an appendix as references for clients, who may be interested in reading them if they have expertise in information security.

4.4 Prioritization Models

This stage of the SQUARE methodology focused on the prioritization of safety and security requirements. To prioritize the safety and security requirements, the SQUARE team utilized the misuse cases and their categories to determine which misuse cases were of utmost importance to ensure the survivability of the Asset Management System. The Acme Company was also asked by the SQUARE team to prioritize the misuse cases based on their security and safety goals (Step 2) for the Asset Management System.

By evaluating each misuse case and its attributes, the SQUARE team was able to prioritize which vulnerabilities and misuses were detrimental to the Asset Management System's normal operations. Using this prioritization approach, it allowed the SQUARE team to provide the necessary security requirements and recommendations to Acme for securing future releases of the Asset Management System.

4.4.1 Methodology

The SQUARE team discussed the prioritization of misuse cases in order to provide the client useful recommendations based on their prioritized safety and security goals. The team ranked each misuse case as a high, medium, or low priority based on its effect on the overall system. A spreadsheet was created for evaluation of each misuse case by averaging ranks from each member of the SQUARE team. Priority levels were based on a 10 point scale with the following classification: 1-3 = Low, 4-6 = Medium, and 7-10 = High. The SQUARE team calculated the average of all numeric ranks to determine the misuse case priority level (Figure 15). The team sought input from the client for their priorities. The average of the SQUARE team was compared against the average of the client. When there were differences in ranks, the team chose to adopt the priority levels of the client because they have better understanding of their system. This process resulted in the identification of twelve misuse cases with high priority. These twelve misuse cases were then incorporated for further research (those with medium and low priority were not addressed). By determining high-priority misuse cases, the SQUARE team was able to provide Acme the necessary security and policy recommendations for future implementations of the Asset Management System. Figure 15 shows the rankings of each of the team members, and their average.

Misuse Case #	TM 1	TM 2	TM 3	TM 4	TM 5	TM 6	TM 7	TM 8	Average	Priority
MC-01	5	7	10	5	6	7	10	5	6.88	Medium
MC-02	8	7	5	8	7	6	8	2	6.38	Medium
MC-03	10	10	7	5	7	6	6	4	6.88	Medium
MC-04	10	8	8	10	10	9	10	2	8.38	High
MC-05	5	6	3	7	8	6	10	7	6.50	Medium
MC-06	10	8	4	8	9	6	10	6	7.63	High
MC-07	6	7	7	8	6	5	7	3	6.13	Medium
MC-08	10	9	7	9	7	6	7	5	7.50	High
MC-09	10	10	10	9	6	8	10	6	8.63	High
MC-10	8	9	8	7	10	8	8	6	8.00	High
MC-11	3	4	6	7	7	5	7	2	5.13	Medium
MC-12	10	10	5	9	10	6	10	8	8.50	High
MC-13	8	9	7	8	10	6	7	6	7.63	High
MC-14	6	4	3	7	10	3	7	2	5.25	Medium
MC-15	6	4	3	6	9	3	5	2	4.75	Medium
MC-16	3	6	5	8	9	5	5	6	5.88	Medium
MC-17	10	9	8	10	10	8	8	8	8.88	High
MC-18	3	6	5	7	7	6	5	5	5.50	Medium
MC-19	5	7	8	9	9	4	5	5	6.50	Medium
MC-20	6	8	7	9	10	8	5	6	7.38	High
MC-21	10	10	10	8	10	8	10	3	8.63	High
MC-22	10	9	8	8	10	5	10	4	8.00	High

Figure 15: Misuse Case Team Priorities

4.4.2 Client Feedback

The SQUARE team incorporated the client's prioritization into the team's rankings. This provided a way to analyze the validity of the prioritizations performed by the team versus those prioritized by the client. Table 7 is the table that the client returned with their prioritization of misuse cases.

Table 7:   Client Prioritization Table

Name	Misuse Case #	Priority
Unauthorized logon on the Windows 2003 server	**MC-01**	High
Sys Admin gains access to system data	**MC-02**	Medium
Users gain Sys Admin rights on the Windows 2003 server (Elevation of Privilege)	**MC-03**	High
Sys Admin deletes critical system configurations on the Windows 2003 server	**MC-04**	High
Sys Admin creates holes in the system configurations on the Windows 2003 server	**MC-05**	Medium
User deletes critical data from the AMS system	**MC-06**	High
User falsifies system data	**MC-07**	Medium
Access system data through developmental machines	**MC-08**	High
Access system data directly to/from database	**MC-09**	Medium
Steal user credential information through developmental machines	**MC-10**	High
User sees data that he or she should not see from his or her workstation	**MC-11**	Medium

Name	Misuse Case #	Priority
Malicious user replays attack in the same browser to assume the identity of another user	**MC-12**	Medium
Malicious users tap communications channel between workstations and servers	**MC-13**	High
Malicious users gain access to sensitive data via saved Excel export files on victim's machine	**MC-14**	Medium
Malicious users install malicious programs that can tap into Excel's memory to steal exported data	**MC-15**	Medium
Input validation attack	**MC-16**	High
Infect server with virus/worms	**MC-17**	High
User gains access to the system using spoofed identities	**MC-18**	Medium
Information gathering/network eavesdropping	**MC-19**	Medium
Brute force attacks: password cracking/credential theft	**MC-20**	High
A user disrupts services (application, software, hardware, and network) in the network, which causes system unavailability/downtime	**MC-21**	High
Execute malicious code	**MC-22**	High

Acme's feedback was then compared to the SQUARE team's average prioritization ranks, as displayed in the figure below (Figure 16). For the misuse cases that the SQUARE team and the client had different opinions on, the SQUARE team decided to adopt the client's priorities.

Misuse Case #	Average	SQUARE Team Priority	Client Priority
MC-01	6.88	Medium	High
MC-02	6.38	Medium	Medium
MC-03	6.88	Medium	High
MC-04	8.38	High	High
MC-05	6.50	Medium	Medium
MC-06	7.63	High	High
MC-07	6.13	Medium	Medium
MC-08	7.50	High	High
MC-09	8.63	High	Medium
MC-10	8.00	High	High
MC-11	5.13	Medium	Medium
MC-12	8.50	High	Medium
MC-13	7.63	High	High
MC-14	5.25	Medium	Medium
MC-15	4.75	Medium	Medium
MC-16	5.88	Medium	High
MC-17	8.88	High	High
MC-18	5.50	Medium	Medium
MC-19	6.50	Medium	Medium
MC-20	7.38	High	High
MC-21	8.63	High	High
MC-22	8.00	High	High

Figure 16: SQUARE Team Versus Client Priorities

4.4.3 Recommendation

The prioritization of misuse cases provided an effective way to assess the client's system. Narrowing down to the most vital and most impacting misuse cases allows for the creation of strategies to provide a strong set of security and policy recommendations. The comparison of the SQUARE team's and the client's priority levels was helpful in determining the overall accuracy of the SQUARE team's expertise and the validity of the methodology.

There were several other prioritization models that were researched but barely pursued during this phase. These included applying the Analytical Hierarchal Process (AHP) for deriving value functions and applying direct utility assessments. These models basically derived quantitative scores based on qualitative scores for the misuse cases. In the AHP example, categories from each misuse case were selected (access right levels, security attributes affected, and sophistication level) and analyzed by pair wise comparisons, resulting in a final set of prioritized misuse cases.

5 Steps 7-9: Architectural and Policy Requirements

5.1 Categorizing and Detailing Recommendations

This stage of the SQUARE methodology focused on categorizing and detailing the architectural and policy requirements. The purpose of this step was to assist Acme in securing their Asset Management System. In order to provide the necessary technical recommendations to Acme, the SQUARE team researched all possible technical remedies and system hardening techniques based on the high-level priority misuse cases. Using this process, the SQUARE team was able to provide all the necessary step-by-step security and technical implementations in order to harden the core components of the Asset Management System (i.e., Sybase Database, ARCHIBUS, Microsoft Windows Server 2003, etc.).

5.1.1 Methodology

The process of categorizing the security and safety requirements involved two steps. First, the SQUARE team researched all possible system hardening and threat prevention techniques that applied to high-level misuse cases. Second, the SQUARE team determined which relevant techniques could be applied to the existing Asset Management System architecture. In order to provide Acme with a complete remedy for each high-level misuse case, the SQUARE team provided a step-by-step hardening solution. The architectural and policy recommendations documented in a misuse cases step were also used as the focus for security requirements. Major Web sites such as Microsoft, Sybase, ARCHIBUS, and CERT were utilized to find detailed methods for hardening vulnerable Asset Management System components. The results varied depending on the availability of researched information and technology. Due to lack of time, safety requirements were not produced for Acme Company. This process resulted in a comprehensive Security Requirements document that focused on technical recommendations such as hardening techniques for Acme to implement in future releases of the Asset Management System. Table 8 gives a description of the attributes in the Security Requirements document. Please refer to Appendix H for the complete document.

Table 8:   Security Requirements Documentation Template

Goal(s)	*The goal(s) refers to higher level objectives that the client desires to achieve and/or wishes to implement.*
Requirement(s)	*The security requirement attempts to narrow down the goal into a rule(s) or regulation(s) containing security issues that may affect the system. Requirements specify the overall security and protection of the data and system. These are numbered based on category (AC: Access Control, EN: Encryption, AU: Auditing, PV: Privacy, AN: Authentication, SU: Survivability, DC: Disaster Control, and UA: Unauthorized Attack).*
Category	*One of eight categories (Access Control ,Encryption, Auditing, Privacy, Authentication, Survivability, Disaster Control and Unauthorized Attack)*
Number	*Architectural or policy recommendation number in reference to each specific misuse case (e.g., AR-01,PR-01)*
Misuse Case	*Reference to all the misuse cases that apply to the specific architectural or policy recommendation (e.g., MC-01, MC-03, MC-04, MC-06, MC-08, MC-10, MC-13, MC-16, MC-17, MC-20, MC-21, MC-22)*
Implementation Choices	*Implementation choices identifies methods in which the client can achieve their goals and requirements through either their existing system technologies or through other technologies in the market.*

After all the requirements were completed, the architectural and policy recommendations were grouped into eight categories: Access Control, Encryption, Auditing, Privacy, Authentication, Survivability, Disaster Control, and Unauthorized Attack. This was done to aid the client in resolving a specific category of misuse cases. For example, if Acme was concerned about dealing with the survivability of the system, the architectural and policy recommendations related to that category could be implemented. Appendix I shows the eight categories and their related architectural and policy recommendations.

Flow diagrams were then drawn based on the Security Requirements document and the categories for easier traceability. This would enable the client to follow the logical structure of implementation from the high-level goals and back again to ensure that every requirement was implemented and that no extraneous functionality was added. Figure 17 below shows an example of a flow diagram. For the complete set of flow diagrams, please refer to Appendix J.
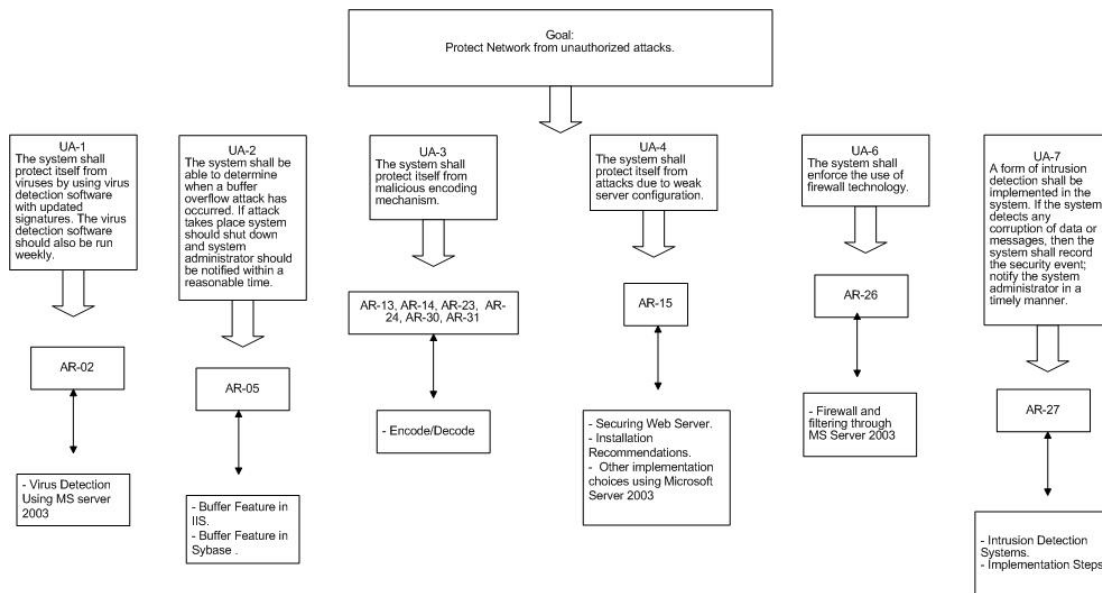
Figure 17: Flow Diagram Example

This step was started and completed without any interaction with the client, and took approximately a month and a half to complete and document. Given some more time, additional detailed architectural and policy recommendations could have been researched and documented.

5.1.2 Client Feedback

Given that categorizing and detailing the recommendations occurred during the final stages of the project, the client did not have the opportunity to give feedback on the process and/or deliverable.

5.1.3 Recommendations

The overall step required a substantial amount of time and effort to find specific implementation choices. The most challenging part was finding system-specific security requirements, given the complexity of the system interactions. Most of the time was spent conducting research for specific implementation methods and writing and revising the security requirements and goals. Therefore, it is recommended that this process be started as soon as the high-level architectural and policy recommendations are set in place. Research was mostly conducted through Internet resources on the vendor sites of the system components used. The most helpful resource in this process was the Microsoft Web site, where most implementation recommendations were taken from. In regards to future work, safety requirements (which were not researched due to time constraints) would be worth investing effort in.

5.2 Budgeting and Analysis

In an ideal situation, all the misuse cases that were prioritized as high (and possibly the mediums and the lows) would be addressed and resolved to protect Acme's Asset Management System. However, due to the limited amount of resources available, only a certain subset of those misuse cases and, inherently, architectural and policy recommendations should be selected. To optimize this subset, a mathematical model was formulated to solve for the best combination of misuse cases. This resulted in a table that referenced which misuse cases were to be addressed based on the budget available.

5.2.1 Methodology

The client was asked to estimate the total yearly costs of implementation for each of the architectural and policy recommendations (only for those that were related to the high-priority misuse cases) in addition to the expected losses of each high-level misuse case given the probability that it was exploited. This was done by first categorizing the different roles of personnel needed to implement the recommendation or resolve the misuse case and assigning an hourly rate for each. Table 9 below shows the hourly rate of each employee title given by the client.

Table 9:   Hourly Rates of Employees

Title	In-House $/hour
IT/Program Manager	50
Database Administrator	44
System Administrator	32
ARCHIBUS Administrator	24
Help Desk Person	18
Programmer	17

Then, each of the recommendations and losses was assigned a category of threat (see Table 10) and had its costs calculated according to specific criteria. Following is a description of the criteria required for each:

Architectural Recommendation Costs

– Category of Threat: Set of related misuses and attacks that pose threat(s) to the organization.
– Implementation Cost: Cost needed to implement (or configure) an architectural recommendation. Could include training costs associated with implementation. Usually a one-time fee expressed in dollars.

- Maintenance Cost: Cost needed to maintain an architectural recommendation after implementation. Includes time spent on recommendation. It is expressed in dollars per year.
- Software Cost: Cost of any software that needs to be purchased, installed, and/or configured in order to implement an architectural recommendation. Usually a one-time fee expressed in dollars.
- Hardware Cost: Cost of any hardware that needs to be purchased, installed, and/or configured in order to implement an architectural recommendation. Usually a one-time fee expressed in dollars.

Policy Recommendation Costs

- Category of Threat: Set of related misuses and attacks that pose threat(s) to the organization.
- Training Cost: Cost needed to educate and train users in the organization about how to correctly implement and enforce a policy recommendation. Could also include training material costs (documents, manuals, etc.) and any other follow-up training sessions needed. Usually a one-time fee expressed in dollars.
- Enforcement Cost: Cost needed to enforce a policy recommendation after implementation. Includes cost of time spent on enforcing recommendation. It is expressed in dollars per year.
- Other Costs: Costs that are specific to the policy recommendation and do not fall under training or enforcement. Could include cost of additional hardware or software. Could be expressed either in dollars or dollars per year, depending on the type.

Misuse Case Losses

- Fixing Cost: Cost needed to fix the result of a misuse case being exploited. Could include costs of external teams that are hired to solve the problem. Expressed in dollars.
- Productivity Loss: Cost of lost productivity when the system or part of the system is non-functional or jeopardized as a result of the misuse case exploitations. Expressed in dollars.
- Other Losses: Cost of other losses that do not fall under any of the other categories and are particular to the specific misuse case. Expressed in dollars.

Some of the costs were calculated as a value derived from the number of hours expected to be spent by a specific employee multiplied by the corresponding hourly rate. Others were estimated based on the client's experience and/or market average. These were all summed up to reflect the total yearly costs per recommendation.

The misuse case losses were calculated on a per incident basis rather than a yearly one, so estimates of misuse case frequencies per year were needed. This was done by assigning values to the threat categories, which in turn were reflected in the misuse cases themselves,

and the total yearly losses were then derived accordingly. Table 10 shows the yearly frequencies assigned by the client.

Table 10:   Threat Categories and Frequencies

Category of Threat	Abbreviation	Frequency (per year)
Active Wiretapping/Network Eavesdropping	W	3
Denial of Service	D	3
Sabotage of Data	S	2
System Penetration	P	3
Theft of Proprietary Information	T	2
Unauthorized Access by Insiders	U	10
Virus	V	15

For a complete reference of the architectural recommendation costs, policy recommendation costs, and misuse case losses, please refer to Appendices K, L, and M respectively.

5.2.1.1 Mathematical Model

From all the total yearly costs and losses, a mathematical model was formulated to optimize the selection of a certain set of misuse cases based on a given budget. The following assumptions were made regarding the misuse cases, architectural recommendations, and policy recommendations:

- Each misuse case is either resolved (i.e., the misuse case will be prevented) or not resolved (i.e., the misuse case will be exploited)—no partial misuse case prevention/ exploitation exists.

- The cost of resolving a misuse case is the sum of all the policy recommendations (PRs) and architectural recommendations (ARs) together (yearly costs calculated).

- All the related ARs and PRs are needed to resolve a specific misuse case. If only one is missing, the misuse case would be considered as unresolved.

- If a misuse case is not resolved, a loss in yearly dollars is incurred.

Based on that, the following were defined:[1]

1. Two vectors of binary variables:

    $X_i = 1$ if recommendation i is implemented and 0 if it is not, i = 1, 2, … P, P+1, …M where there are P architectural recommendations and M – P policy recommendations.

    $Y_j = 1$ if misuse case j is not resolved and 0 if it is; j = 1, 2, … N

---

[1]   Caulkins, Jon. "Re: Information Security IP Formulation Discussed Yesterday" [email to Hasan Osman], [online], July 14, 2004.

2. Two associated sets of constants:

   c_i = cost of taking action i

   w_j = losses incurred if misuse case j is not resolved

3. A many-to-many relationship:

   S_j = set of changes that must be implemented to solve misuse case j

   For example, if one had to implement recommendations X_2, X_5, and X_{19} to solve misuse case j = 6, then $S_6 = \{2, 5, 19\}$.

4. A $|\ |$ operator to denote the cardinality of a set, e.g., $|S_6| = 3$.

5. A yearly budget B available.

The following formulas define the Integer Program objective function and constraints:

$$Min \sum_{i=1}^{M} c_i\, X_i + \sum_{j=1}^{N} w_j Y_j$$

Subject to:

$$Y_j + \frac{\sum\limits_{i \in S_j} X_i}{|S_j|} \geq 1$$

$$\sum_{i=1}^{M} c_i\, X_i \leq B$$

The integer program was then inputted into an Excel spreadsheet, and the model was executed on yearly budgets ranging from $0 to $200,000 in $5,000 increments. For each budget, a related set of misuse cases were selected, and their total cost was calculated. Table 11 shows the results:

Table 11: Selected Misuse Cases Based on Budget

Total Budget ($/Year)	Corresponding MCs	Total Cost of Misuse Cases ($/Year)
0	-	0
5,000	-	0
10,000	MC-20	9,000
15,000	MC-20	9,000
20,000	MC-20	9,000

Total Budget ($/Year)	Corresponding MCs	Total Cost of Misuse Cases ($/Year)
25000	MC-20	9,000
30,000	MC-20	9,000
35,000	MC-20	9,000
40,000	MC-20	9,000
45,000	MC-04, MC-16, MC-20, MC-22	43,907
50000	MC-04, MC-16, MC-20, MC-22	43,907
55,000	MC-04, MC-13, MC-16, MC-20, MC-22	53,689
60,000	MC-01, MC-03, MC-04, MC-06, MC-08, MC-16, MC-20, MC-22	57,739
65,000	MC-01, MC-03, MC-04, MC-06, MC-08, MC-13, MC-16, MC-20, MC-22	63,873
70,000	MC-01, MC-03, MC-04, MC-06, MC-08, MC-10, MC-13, MC-16, MC-20, MC-22	67,673
75,000	MC-16, MC-17, MC-22	73,474
80,000	MC-16, MC-17, MC-20, MC-22	77,674
85,000	MC-04, MC-16, MC-17, MC-20, MC-22	81,526
90,000	MC-04, MC-13, MC-16, MC-17, MC-20, MC-22	87,792
95,000	MC-04, MC-13, MC-16, MC-17, MC-20, MC-22	87,792
100,000	MC-01, MC-03, MC-04, MC-06, MC-08, MC-13, MC-16, MC-17, MC-20, MC-22	97,976
105,000	MC-01, MC-03, MC-04, MC-06, MC-08, MC-10, MC-13, MC-16, MC-17, MC-20, MC-22	101,776
1E+11	MC-01, MC-03, MC-04, MC-06, MC-08, MC-10, MC-13, MC-16, MC-17, MC-20, MC-22	101,776

A graph was then plotted showing the cost of resolved misuse cases versus the budget allocated (see Figure 18).

Cost of Resolved Misuse Cases Vs. Budget
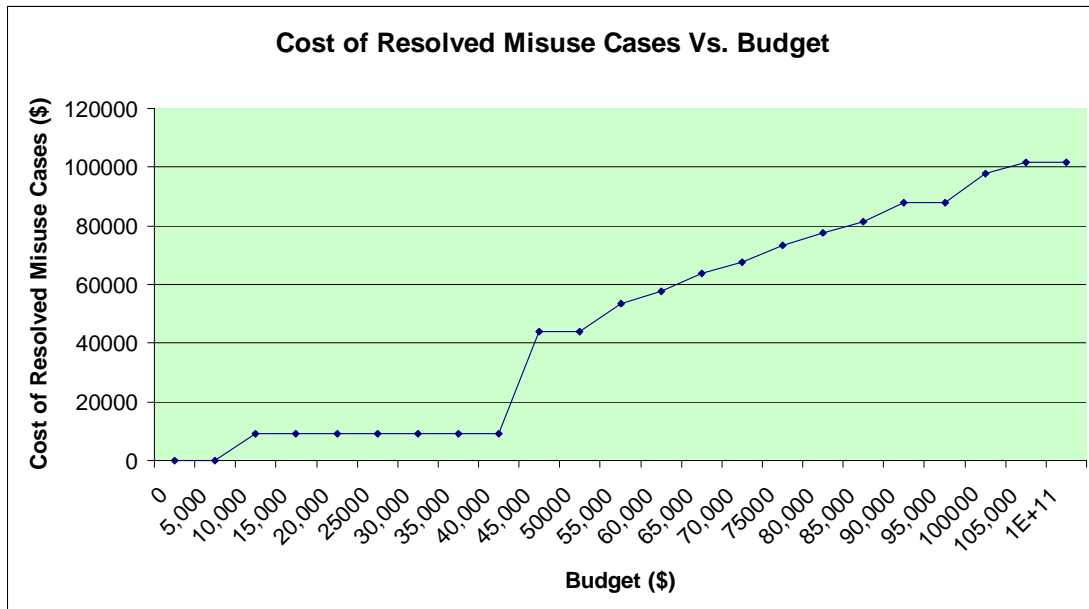
Figure 18: Plot of Misuse Case Cost Versus Budget

The results basically showed that

- At least $9,000 is needed to make it feasible to resolve one misuse case (MC-20).

- Any budget between $9,000 and $43,000 would still result in only MC-20 being resolved. An amount of around $44,000 is needed to resolve additional misuse cases.

- Even with an unlimited budget, one of the misuse cases (MC-21) would not be selected as an option to be resolved given that its cost to implement is much more than the losses it would incur if left unattended.

- A sum of around $102,000 would be enough to resolve 11 out of the total 12 misuse cases.

Based on how much the client would be willing to spend, the related misuse cases could then be resolved by implementing the corresponding architectural and policy recommendations.

5.2.2 Client Feedback

During this whole process, the client was quite challenged in coming up with numbers. For the misuse case losses in particular, the numbers were literally guessed. This was due to the client's inexperience in calculating how much time or effort would be needed in fixing a problem that never occurred. Even calculating the architectural and policy recommendation costs had to be estimated within a relatively large margin of error due to the complexity of approximating implementation procedures.

The process of calculating the costs and losses took around 10 days, with only a few clarifications that were needed to be addressed for the client through phone and email. The

mathematical model required a bit more time (around two and a half weeks) to be finalized, after a substantial amount of research time was invested. The formalization of the model did not include any interaction with the client, given that there was no direct need.

5.2.3 Recommendations

The client was helpful and proactive in supplying the team with data on costs and losses, despite the difficulty of generating accurate figures. The team originally tried to minimize the categories of the costs that needed to be filled out to minimize the effort required by the client. However, the client ended up adding some more information that turned out to be beneficial. For example, the enforcement costs of the policy recommendations required an estimate of the number as one lump sum yearly cost, but the client added in the amount of hours required by each employee per month, and multiplied that by the hourly rate of the corresponding title. Apparently, it was easier for the client to estimate the costs and losses through hourly rates rather than lump sum figures. This should be taken into consideration when dealing with future clients.

For the misuse cases, potential work lies in analyzing any dependencies within the architectural and policy recommendations (which were assumed to be independent for simplification purposes). This would also reflect on the analysis of compensating for partial misuse case resolution – which was also not taken into account for simplification reasons.

For future applications of this step, it would also be recommended to use the categories and templates included along with this report, which were quite helpful. Nevertheless, more research could also be done for other methods in calculating costs and losses, depending on the system analyzed.

6 Conclusion

The original nine steps of the SQUARE methodology were lumped into four based on how the process was approached; however, in any other future applications of the methodology, lumping those steps might not be applicable, and should be taken into consideration accordingly. Moreover, some steps were applied without any formal process defined. Elicitation techniques, for example, were informally touched on, given the nature of interaction with the client.

Although the company had around 1,000 employees nationwide, the SQUARE team primarily dealt with only two (technical lead and assistant), and most of the time, only one. This is another point that needs to be taken into account for future applications. However, this actually turned out to be an advantage in the sense that not too much time was wasted on internal processes or meetings with the client, which would have substantially increased the time frame. Another important point was the efficiency with which document management was conducted. It was ideal to have one person from each side communicating all the changes and updates, which reduced error and enhanced the document control process.

The team viewed the end result of this report as a collection of recommendations for each step of the process followed, in addition to a description of the results and their intended purpose. The SQUARE methodology is still under review by the Software Engineering Institute's Networked Systems Survivability Program. It demonstrates great potential for industry-wide adoption for developing secure applications and systems.

Appendix A   Definitions

The following are the definitions of terms agreed on by the client and the SQUARE team. Resources were used from the SEI, the CERT/CC, and other renowned sources.

access control	Access control ensures that resources are only granted to those users who are entitled to them [SANS 03a].
access control list	A table that tells a computer operating system which access rights or explicit denials each user has to a particular system object, such as a file directory or individual file [TechTarget 03a].
antivirus software	A program that searches hard drives and floppy disks for any known or potential viruses [TechTarget 03a].
artifact	The remnants of an intruder attack or incident activity. These could be software used by intruder(s), a collection of tools, malicious code, logs, files, output from tools, or the status of a system after an attack or intrusion [West-Brown 03].
asset	A critical valuable that a company owns and wants to secure.
attack	An action conducted by an adversary, the attacker, on a potential victim. A set of events that an observer believes to have information assurance consequences on some entity, the target of the attack [Ellison 03].
auditing	The information gathering and analysis of assets to ensure such things as policy compliance and security from vulnerabilities [SANS 03a].
authentication	The process of determining whether someone or something is, in fact, who or what it is declared to be [TechTarget 03b].

availability	The property of a system or a system resource being accessible and usable upon demand by an authorized system entity, according to performance specifications for the system; i.e., a system is available if it provides services according to the system design whenever users request them [Allen 99].
back door	An element in a system that allows access by bypassing access controls [Howard 97].
breach	Any intentional event in which an intruder gains access that compromises the confidentiality, integrity, or availability of computers, networks, or the data residing on them [CERT/CC 04].
brute force	A cryptanalysis technique or other kind of attack method involving an exhaustive procedure that tries all possibilities, one by one [SANS 03a].
buffer overflow	A buffer overflow occurs when a program or process tries to store more data in a buffer (temporary data storage area) than it was intended to hold. Since buffers are created to contain a finite amount of data, the extra information—which has to go somewhere—can overflow into adjacent buffers, corrupting or overwriting the valid data held in them [SANS 03a].
cache cramming	The technique of tricking a browser to run cached Java code from the local disk instead of the Internet zone, so it runs with less restrictive permissions [SANS 03a].
cache poisoning	Malicious or misleading data from a remote name server is saved [cached] by another name server. Typically used with Domain Name System (DNS) cache poisoning attacks [SANS 03a].
confidentiality	The property that information is not made available or disclosed to unauthorized individuals, entities, or processes (i.e., to any unauthorized system entity) [SANS 03a].
control	An action, device, procedure, or technique that removes or reduces a vulnerability.
corruption	A threat action that undesirably alters system operation by adversely modifying system functions or data [SANS 03a].
cracker	Someone who breaks into someone else's computer system, often on a network; bypasses passwords or licenses in computer programs; or in other ways intentionally breaches computer security [TechTarget 04b].

denial-of-service (DoS) attack	A form of attacking another computer or company by sending millions of requests every second, causing the network to slow down, cause errors, or shut down [Computer Hope 04].
disaster recovery plan	A disaster recovery plan (DRP)—sometimes referred to as a business continuity plan (BCP) or business process contingency plan (BPCP)—describes how an organization is to deal with potential disasters [TechTarget 04c].
disclosure	The dissemination of information to anyone who is not authorized to access that information [Alberts 03].
disgruntled employee	A person in an organization who deliberately abuses or misuses computer systems and their information [Alberts 03].
downtime	The amount of time a system is down in a given period. This will include crashes and system problems as well as scheduled maintenance work [RUsecure 04b].
disruption	A circumstance or event that interrupts or prevents the correct operation of system services and functions [Alberts 03].
encryption	Cryptographic transformation of data (called "plaintext") into a form (called "cipher text") that conceals the data's original meaning to prevent it from being known or used [SANS 03a].
espionage	The act or practice of spying or of using spies to obtain secret information, as about another government or a business competitor [Dictionary.com 04b].
essential services	Services to users of a system that must be provided even in the presence of intrusion, failure, or accident [Ellison 97].
exposure	Same as **disclosure**.
fabrication	Same as **masquerade**.
fault line attacks	Fault line attacks use weaknesses between interfaces of systems to exploit gaps in coverage [SANS 03a].
fault tolerance	Describes a computer system or component designed so that, in the event that a component fails, a backup component or procedure can immediately take its place with no loss of service. Fault tolerance can be provided with software, or embedded in hardware, or provided by some combination [TechTarget 03d].

firewall	A system designed to prevent unauthorized access to or from a private network. Firewalls can be implemented in both hardware and software, or a combination of both [Webopedia 04a].
hacker	An individual who breaks into computers primarily for the challenge and status of obtaining access [Howard 97].
honey pot	Programs that simulate one or more network services designated on a computer's ports. An attacker assumes that vulnerable services that can be used to break into the machine are being run. A honey pot can be used to log access attempts to those ports, including the attacker's keystrokes. This can provide advanced warning of a more concerted attack [SANS 03a].
HTTP header manipulation	HTTP requests and responses send information in the HTTP headers. HTTP headers are a series of lines containing a name/value pair used to pass information such as the host, referrer, user agent, etc. HTTP headers can be manipulated to cause SQL injection or cross-site scripting errors [ASI 04].
impact	The negative effect of an attack on a victim system by an attacker [Allen 99].
incident	An incident is an adverse network event in an information system or network or the threat of the occurrence of such an event [SANS 03a].
incident handling	An action plan for dealing with intrusions, cyber theft, denial of service, fire, floods, and other security-related events [SANS 03a].
insider threat	The threat that authorized personnel of an organization will act counter to the organization's security and interest, especially for the purposes of sabotage and espionage [NIPC 02].
integrity	For systems, the quality that a system has when it can perform its intended function in a unimpaired manner, free from deliberate or inadvertent unauthorized manipulation. For data, the property that data has not been changed, destroyed, or lost in an unauthorized or accidental manner [Allen 99].
interception	Access to an asset gained by an unauthorized party [Pfleeger 03].

interruption	An event that causes an asset of a system to be destroyed or become unavailable or unusable [Howard 97].
intrusion	An attack on a network for the purpose of gaining access to or destroying privileged information or disrupting services to legitimate users [Ellison 03].
intrusion detection system	A combination of hardware and software that monitors and collects system and network information and analyzes it to determine if an attack or an intrusion has occurred. Some ID systems can automatically respond to an intrusion [Allen 99].
intrusion prevention system	A system used to actively drop packets of data or disconnect connections that contain unauthorized data. Intrusion prevention technology is also commonly an extension of intrusion detection technology [Wikipedia 04].
liability	The responsibility of someone for damage or loss [West-Brown 03].
luring attack	A type of elevation of privilege attack where the attacker "lures" a more highly privileged component to do something on his or her behalf. The most straightforward technique is to convince the target to run the attacker's code in a more privileged security context [Brown 05].
malware	Programming or files that are developed for the purpose of doing harm. Thus, malware includes computer viruses, worms, and Trojan horses [Webopedia 04b].
man-in-the-middle attack	An attack in which the attacker is able to read, and possibly modify at will, messages between two parties without letting either party know that they have been attacked. The attacker must be able to observe and intercept messages going between the two victims [Farlex 04].
masquerade	Aims to fool other machines on the network into accepting the imposter as an original, either to lure the other machines into sending it data or to allow it to alter data [Howard 98].
modification	Situation in which an unauthorized party not only gains access to, but tampers with an asset [Howard 97].
non-essential services	Services to users of a system that can be temporarily suspended to permit delivery of essential services while the system is dealing with intrusions and compromises [Ellison 97].

non-repudiation	The goal of non-repudiation is to prove that a message has been sent and received [SSI 03].
patch	A small update released by a software manufacturer to fix bugs in an existing program [SANS 03a].
patching	The process of updating software to a new version that fixes bugs in a previous version [SANS 03a].
penetration	Intrusion, trespassing, or unauthorized entry into a system [RUsecure 04c].
penetration testing	The execution of a testing plan, the sole purpose of which is to attempt to hack into a system using known tools and techniques [RUsecure 04d].
physical security	Security measures taken to protect systems, buildings, and related supporting infrastructure against threats associated with their physical environment [Guttman 95].
port scanning	The act of systematically scanning a computer's ports [Webopedia 04c].
privacy	The quality or condition of being secluded from the presence or view of others [Dictionary.com 04c].
procedure	The implementation of a policy in the forms of workflows, orders, or mechanisms [West-Brown 03].
recognition	The capability of a system to recognize attacks or the probing that precedes attacks [Ellison 03].
recovery	A system's ability to restore services after an intrusion has occurred. Recovery also contributes to a system's ability to maintain essential services during intrusion [Ellison 03].
replay attack	The interception of communications, such as an authentication communication, and subsequent impersonation of the sender by retransmitting the intercepted communication [FFIEC 04].
resilience	The ability of a computer or system to both withstand a range of load fluctuations and also remain stable under continuous and/or adverse conditions [RUsecure 04e].
resistance	Capability of a system to resist attacks [Ellison 03].
risk	The product of the level of threat with the level of vulnerability. It establishes the likelihood of a successful attack [SANS 03a].

risk assessment	The process by which risks are identified and the impact of those risks determined [SANS 03a].
security policy	A policy that addresses security issues [West-Brown 03].
script kiddies	The more immature but unfortunately often just as dangerous exploiter of security lapses on the Internet. The typical script kiddy uses existing and frequently well-known and easy-to-find techniques and programs or scripts to search for and exploit weaknesses in other computers on the Internet—often randomly and with little regard or perhaps even understanding of the potentially harmful consequences [TechTarget 03f].
spoof	The term is used to describe a variety of ways in which hardware and software can be fooled. IP spoofing, for example, involves trickery that makes a message appear as if it came from an authorized IP address [Webopedia 04d].
SQL injection	A type of input validation attack specific to database-driven applications where SQL code is inserted into application queries to manipulate the database [SANS 03b].
stakeholder	Anyone who is a direct user, indirect user, manager of users, senior manager, operations staff member, support (help desk) staff member, developer working on other systems that integrate or interact with the one under development, or maintenance professionals potentially affected by the development and/or deployment of a software project [Ambler 04].
stealthing	A term that refers to approaches used by malicious code to conceal its presence on an infected system [SANS 03a].
survivability	The capability of a system to complete its mission in a timely manner, even if significant portions are compromised by attack or accident. The system should provide essential services in the presence of successful intrusion and recover compromised services in a timely manner after intrusion occurs [Mead 03].
target	The object of an attack, especially host, computer, network, system, site, person, organization, nation, company, government, or other group [Allen 99].
threat	A potential for violation of security, which exists when there is a circumstance, capability, action, or event that could breach security and cause harm [SANS 03a].

threat assessment	The identification of the types of threats that an organization might be exposed to [SANS 03a].
threat model	Used to describe a given threat and the harm it could to do a system if it has a vulnerability [SANS 03a].
toolkits	A collection of tools with related purposes or functions, e.g., antivirus toolkit, disk toolkit [RUsecure 04f].
Trojan	A program in which malicious or harmful code is contained inside apparently harmless programming or data in such a way that it can get control and do its chosen form of damage, such as ruining the file allocation table on a hard disk [TechTarget 04e].
trust	Determines which permissions other systems or users have and what actions they can perform on remote machines [SANS 03a].
uptime	Same as **availability**.
victim	That which is the target of an attack. An entity may be a victim of either a successful or unsuccessful attack [SANS 03a].
virus	A hidden, self-replicating section of computer software, usually malicious logic, that propagates by infecting—i.e., inserting a copy of itself into and becoming part of—another program. A virus cannot run by itself; it requires that its host program be run to make it active [SANS 03a].
vulnerability	A condition or weakness in (or absence of) security procedures, technical controls, physical controls, or other controls that could be exploited by a threat [Guttman 95].
worm	A self-replicating virus that does not alter files but resides in active memory and duplicates itself. Worms use parts of an operating system that are automatic and usually invisible to the user. It is common for worms to be noticed only when their uncontrolled replication consumes system resources, slowing or halting other tasks [TechTarget 04g].

Appendix B   Safety and Security Goals

The Safety and Security Goals document outlines Step 2 of the System Quality Requirements Engineering's methodology and presents preliminary findings of the SQUARE team's investigation of the security goals regarding the Asset Management System (hereafter referred to as "AMS") by the Acme Group. The report also outlines suggested security objectives and policies. After a brief, preliminary investigation of the system studied, the SQUARE team offered the following observations and analysis. The team understands that the Acme Group lacks security measures/mechanisms in the current build of the AMS system. The following sections outline stakeholders, business objectives, security goals and policies for AMS, and present the Acme Group with a set of inquiries that would aid in refining the objectives.

Business Objectives

The Asset Management System is a facility management tool that aids companies in planning emergencies properly, ensuring the appropriate response. According to the OFM Asset Management Tool-Business Requirements Report:

> *This tool is not intended to provide canned responses to every possible scenario, but instead provides the means to make informative decisions based on available sources.*[2]

AMS assists the client in evaluating and assessing its key components, assets, and personnel in order to develop an effective response plan/tactic in any given situation. The SQUARE team also expects future versions/builds of the AMS system will maintain overall security to ensure proper functionality of its software components. Hence, incorporating security should work in parallel with the original objectives.

Methodology

In studying the stakeholders of the system, the SQUARE team had to understand both the architecture and network topology of AMS. Given that there is no "typical" setup of the system (due to the fact that the end user more or less defines that), one had to be specifically focused on to outline who the main stakeholders are (who, in turn, are recursively dependant on the network and system architecture interactions). In doing that, the current installation on the Acme Group's systems were chosen as the candidate architecture to be studied and analyzed. Figure 19 and Figure 20 show the typical network topology diagram and system architecture for AMS.

---

[2]   Rectenwald, R. J. "OFM Asset Management Tool-Business Requirements Report." Pittsburgh, PA. Acme Group, March 2002.

TYPICAL NETWORK TOPOLOGY FOR ASSET MANAGEMENT SYSTEM

Servers   Servers   Workstations Workstations   Desktop

Figure 1: A typical network topology diagram for Asset Management System. Installation (As implemented on Acme Group's Network)

Servers   Workstations   Firewall   Desktop   Desktop

DMZ   IIS Server

Router

Internet

Figure 19: Network Topology Diagram

SYSTEM ARCHITECTURE

Windows Server (2003 SE)

EPP Documents   *.asp, *.css, *.js, *.gif,   *.dwg files for   *.sdf files for
                *.jpg files for website   AutoCAD   MapGuide

AMS User W2K Workstation

Microsoft Internet Explorer (Ver 6)

MapGuide Author

Microsoft IIS (Ver 6.0)   MapGuide Ver (6.3)

Figure 2: A typical system architecture diagram for Asset Management System Installation.

Microsoft Office 2000

Sybase Central (Ver 4.1)

AMS Developmental W2K Workstation

SDF Loader Program

AutoCad 2004 w/ Archibus Overlay

Development Tools

Edit Plus

MS MapPoint   MS Word 2000   MenuMaker

Archibus-FM (Ver 14.1)
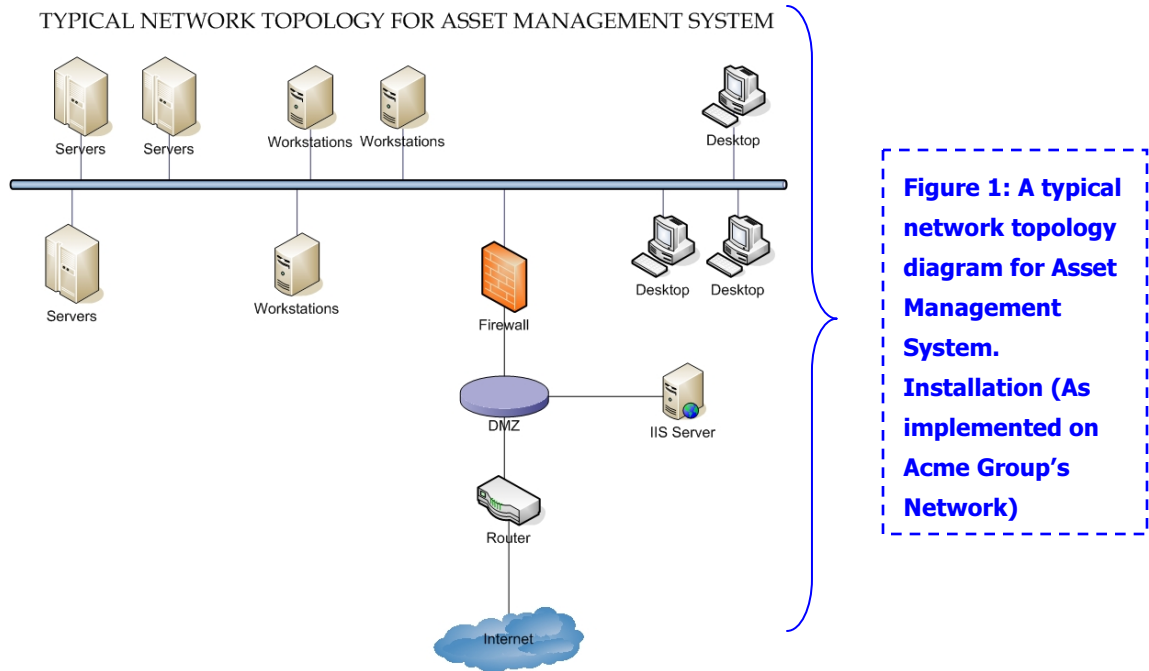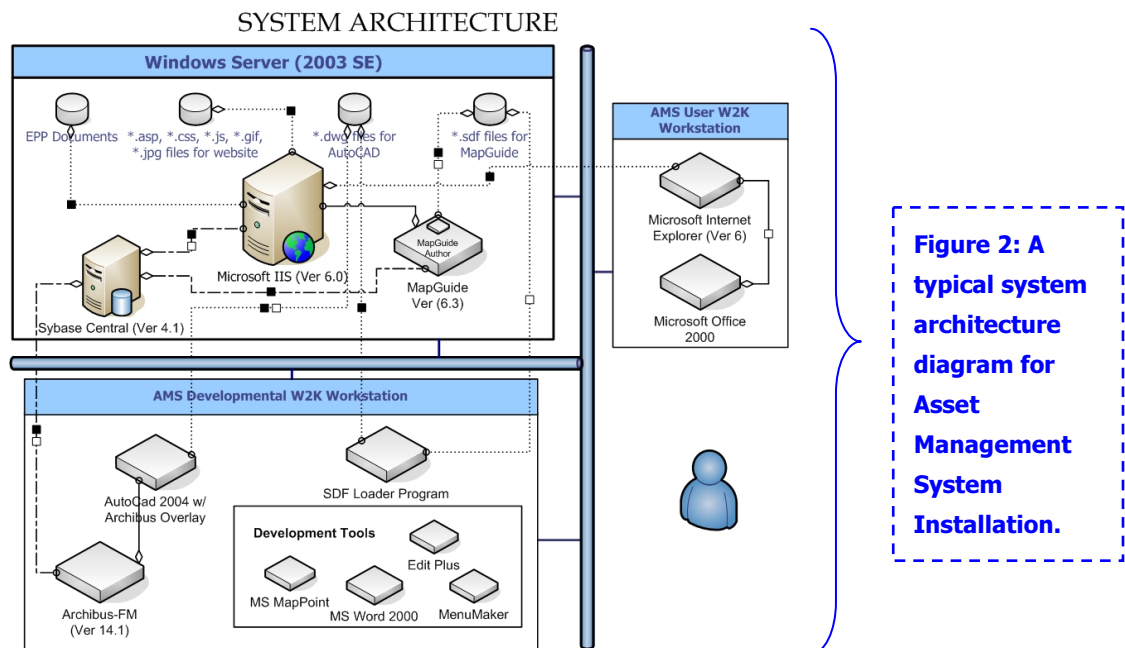
Figure 20: System Architecture Diagram

Stakeholder Analysis

This section analyzes stakeholders of AMS , who are assumed to be the direct users in the preliminary analysis. The following is a breakdown chart of the direct users.

System Administrator

These users will have full control over the Windows 2003 Server and its configuration files. To be granted System Administrator authority of the AMS, the employee must be involved with the maintenance and support of all modules/components, which include:

- Microsoft IIS Server. This includes html files, scripts, etc. in the IIS Server.

- Sybase Central Database.

- MapGuide Ver (6.3). This includes all data that are accessible through MapGuide.

- Event Logs. System Administrator(s) will be able to perform maintenance and review of all event logs stored in the Sybase databases, IIS Server, and Windows 2003 Server.

High Level System User

These users will have read, write, modify, and delete access permissions in the AMS developmental workstations. To be granted high-level access, the employee/personnel must be involved with the maintenance and support of all the modules/components within the developmental workstations, but not to the Windows Server. Components/modules include:

- ARCHIBUS Facility Management. This includes event logs, database entries, and storage.

- Time and Attendance System. Review the inputs of data in the system.

- Facility Drawings and Procedures. Ability to make updates and modifications of facility drawings, policies, and procedures.

- Event Logs. Users will be able to perform maintenance and review of all event logs stored in the Sybase databases.

Medium Level System User

These users will have read access permissions in the AMS system. Users can only access and modify certain "user-only" components of the system. However, these users will not be able modify the core/key components on the actual configuration of the system, but would have modification privileges for attributes.

In reference to Figure 20, the low-authority users will only be able to access the AMS Workstations, and not the AMS Developmental Workstations or the Windows Server.

Low Level System User

These users will only have read access to the system. Users will only be allowed to read data that is intended for public viewing.

In reference to Figure 20, the low-level system users will only have access to the AMS Workstations, and not to the AMS Developmental Workstations or the Windows Server. Please note that the low-level system user accounts/access authority have not been implemented during this review.

In summary, the system administrators and high-level system user authority users are the main stakeholders of the system. If the system malfunctions, the users are responsible for problem resolution.

Security Objectives

These goals are to aid the Acme Group in assessing its existing security standards/policies regarding the Asset Management System software package from a high standpoint. The SQUARE team recommends the following goals for evaluation and implementation on the current system:

Confidentiality

The security goal of confidentiality will help insure that information and resources are accessed only by those who are authorized to use them. Confidentiality is closely related to Access Control in that it is a component of confidentiality. The protection against unauthorized disclosures will guard against malicious coding, hackers/crackers, and accidental disclosures. Confidentiality also involves policy and procedures as well as the implementation of security controls.

Operating system security policies should be defined, enabled, and monitored to automatically control items such as password complexity and expirations, logon rules, idle time rules, etc. to ease the burden for both the users and system administrators.

Corporate policies should be in place to define end-users' responsibilities in maintaining IT security. Penalties for misconduct should be defined and enforced. Personnel should acknowledge reviewing the document by signature upon hire, significant changes are made; or on a periodic basis.

Availability

This goal is to ensure that the Asset Management System is functional and available at all times. This includes the core facility management services, Sybase databases, etc. The SQUARE team assumed that the system and its core components are back upped regularly either in digital mediums (i.e., hard-disks, DVD-ROMs, etc.) or spare servers.

The AMS client should assess the use of the system and determine their disaster recovery needs. All data should be backed-up to tape or disk daily, with the most minimal scenario performed in a repetitive and consistent manner as follows:

- Perform a full backup once a week on servers housing AMS applications and database.

- Perform incremental backups or differential incremental backups the remaining six days.

- Use a different tape or disk each day in a 4-week rotation.

- Perform a full backup once a month and include that media in a 12-month rotation.

- If possible, store all of the backup media in a remote location. At the very least, keep the backup media in a fire safe in an environmentally controlled area. Media in the 12-month rotation should be stored in a remote location.

- Backup administrators should coordinate efforts with database administrators to determine the backup and restoration techniques for the AMS database and logs.

- Backup administrators should be tenacious in the performance, monitoring, and testing of data/system backups and restorations.

Clients of an AMS application classified as critical should determine what other disaster recovery procedures (e.g., hardware redundancy, data mirroring, remote data mirroring, remote disaster recovery facility) are necessary.

All AMS clients should have a designated laptop devoted to being used as an emergency AMS application server. It should be configured with all of the applications necessary to run the AMS as a mobile, standalone version. The laptop should be readily accessible and scheduled to receive daily updates/downloads from the corporate AMS in order to keep it synchronized.

Data Integrity

Integrity of data is absolutely critical in the Asset Management System package. If the underlying information upon which facility managers must make their decisions is corrupted or wrong, the purpose of the package has been defeated. That solidifies that the issue of data backups and checksum integrity verification are of the utmost importance.

At the very least, full/daily backups are retained for 4 weeks; a full backup is performed once a month and retained for 12 months, with one monthly full backup removed once a year on a scheduled month and archived indefinitely.

Monitoring

One security goal is to preserve or enhance the ability to accurately record the activities that take place. When users interact with the system, a complete accounting of all the commands issued as well as the internal transactions of the package should be available. In order for this to occur, Logging Capabilities that are currently in place for the Asset Management System are needed.

Initially, full logging should be maintained for all AMS applications until installation and acceptable performance goals have been reached. If disk space becomes an issue, the log file content and/or retention time may be reduced as long as security breaches are still captured and application messages are at such a level that they can be used to debug errors quickly.

Access Control

Another goal is to ensure that only authorized users of the Asset Management System have access to their specified and permissible resources. The team would like information on what authentication procedures are in place and if the system will allow for remote connectivity. If this holds true, information regarding intact authentication controls and penetration testing insure the access controls are working properly are needed.

The username/password policy of the Windows network should address both password complexity and expiration. The network user name of the person logged into the workstation accessing the AMS is used as the authenticating identity. Offsite access and/or system administration should not bypass Windows authentication and other security measures.

The level of AMS user-access is specified in the afm_users table and checked/referenced in asp pages for different functions. The level of access is dependent on the level of responsibility the user has in the AMS application with respect to security, sensitive data, human resources records, etc. An AMS client project representative who is familiar with both the corporate organization and the application should provide the user-level access. The afm_users table is edited by the on-site ARCHIBUS administrator (defined as High-Level System User) to reflect the access levels of a given network user name.

Maintaining Mission Critical Services

The most important goal is the ability to deliver essential services in the face of attack, failure, or accident. This is dependant upon maintaining necessary system properties in unfavorable environments. First, the need to identify the critical services that must be delivered and proceed to identify the resources that support those services are required. After identification of critical services and resources, the SQUARE team will proceed to implement controls and defensive measures for protection. Therefore, it follows that one important piece of information would be the critical services within the system.

Disaster Recovery

Disaster recovery was previously defined as the process of recovering IT systems from disasters. Another goal is to have a current disaster recovery plan in the event of an emergency or service disruption. The AMS client should have procedures in place to address disaster recovery plans for different levels of applications and degrees of disaster. The plans should address how and when key personnel are contacted along with their duties and responsibilities. These plans should be tested with the results and lessons learned documented in a central location for easy reference.

Code Review

Periodic code review should be performed to ensure script and code confidentiality and integrity. This process verifies that malicious scripts or code have not been inserted into the source code of the software suite. The potential risks of not performing this periodic review are the take down of the program and potential loss of all data.

Not addressed but needs to be, along with policy for installing Windows updates.

Appendix C   Use Cases

Use cases provide an outline of the system's functionality from a user's perspective, with classification of user level privileges by ACLs. It provides a detailed step of the various ways the Asset Management System Software Suite can be accessed.

User Level Definitions

Low-level user	view only
Medium-level user	general Asset Management System user with edit privileges (journal entries, mark-up floor plans for room status)
High-level user	ARCHIBUS administrator at client site (edit database to add users to afm_users table, create links to EP procedures/docs, etc)
System Administrator	IIS configuration, access controls, user accounts, etc.

Precondition for all Asset Management System-based use cases:

> Login

> OS-based

Unknown users are not permitted access to the Asset Management System Web site.

Number	UC-01
Use Case	View Floor Plans
Description	All level of users able to access the Asset Management System will have the ability to view authorized system information per the ACL such as floor plans, damaged areas, employee locator, etc.
Actors	Low-Level User, Medium-Level User, High-Level User, or System Administrator
Assumptions	- System Admin has added viewing privileges to the ACL - System is available - Data entered is correct
Steps	1. User will enter the URL associated with the Asset Management System. 2. User will receive a prompt to log in their user name and password. 3. The system authorizes and authenticates the user, then allowed into the system. 4. The system will allow them to access privileges as specified by the ACL. 5. From here, the user will navigate to Operations/Maintenance. Choose appropriate property and then floor plans.
Variations	Once logged in, the user can also click on the floor plans tab on the right hand side of the Asset Management System main page.
Non-Functional	They will not have edit privileges; view-only privileges will be assigned. If the user attempts to access unauthorized information, the system will display a pop up window stating that the user is not authorized to access this information.
Related Misuse Cases	MC-01, MC-08, MC-11, MC-12, MC-13, MC-14, MC-15, MC-16, MC-17, MC-18, MC-19, MC-20, MC-21, MC-22

Number	UC-02
Use Case	Damage Assessment
Description	The Medium-Level Asset Management System user wants to make changes to the floor plan to indicate damaged areas in the facility.
Actors	Medium-Level User, High-Level User, System Administrator
Assumptions	1. The user has proper edit privileges 2. The data entered is correct 3. The user has proper security privileges
Steps	1. Select Operations Management. 2. Select Building. 3. Select Floor Plans. 4. Select Area Status to view the current condition. 5. Highlight the specific area for damage assessment. 6. From the drop down menu select the status you wish to assign to the room (Damaged, Destroyed, Inventory, Not Usable, Renovation, Construction). 7. Press Go. 8. To continue marking areas select "Floor Plan" and choose another floor. Repeat steps 4-7.
Variations	N/A
Non-Functional	N/A
Related Misuse Cases	MC-01, MC-06, MC-07, MC-08, MC-11, MC-12, MC-13, MC-14, MC-15, MC-16, MC-17, MC-18, MC-19, MC-20, MC-21, MC-22

Number	UC-03
Use Case	Mark Up/Create Floor Plans
Description	Medium-level users and higher will have the ability to access the Asset Management System and editing privileges such as mark up/create floor plans.
Actors	Medium-Level User, High-Level User, or System Administrator
Assumptions	*You must be an ARCHIBUS user* - System Admin has added editing privileges to the ACL - System is available - Data entered is correct
Steps	1. Create an floor plan drawing in AutoCAD. 2. Add assets to the drawing using ARCHIBUS. 3. Link assets to the floor using ARCHIBUS. 4. Run Acme's proprietary application ABC DFR that defines the base drawing.
Variations	N/A
Non-Functional	If the user attempts to access unauthorized information, the system will display a pop up window stating that the user is not authorized to access this information.
Related Misuse Cases	MC-01, MC-06, MC-07, MC-08, MC-11, MC-12, MC-13, MC-14, MC-15, MC-16, MC-17, MC-18, MC-19, MC-20, MC-21, MC-22

Number	UC- 04
Use Case	Find Specialized Employees
Description	The Medium-Level Asset Management System user wants to make changes to the data to indicate damaged areas in the facility.
Actors	Low-Level User, Medium-Level User, or High-Level User
Assumptions	The user has proper security privileges
Steps	*Version 1* 1. Select Facility. 2. Select Personnel Re-Call List.
Variations	*Version 2* 1. Select Facility. 2. Under the "Business Community" heading select Personnel Call List. *Version 3* 1. Select Ad-Hoc Event Management. 2. Select Employee Locator. 3. Select "Set Restriction." 4. Add in filtering Information for query.
Non-Functional	N/A
Related Misuse Cases	MC-01, MC-08, MC-11, MC-12, MC-13, MC-14, MC-15, MC-16, MC-17, MC-18, MC-19, MC-20, MC-21, MC-22

Number	UC-05
Use Case	Journal Entry
Description	Medium-Level users and higher will have the ability to access the Asset Management System and journal entry privileges.
Actors	Medium-Level User, High-Level User, or System Administrator
Assumptions	This assumes that: - System Admin has added viewing privileges to the ACL - System is available - Data entered is correct
Steps	Adding Entry 1. Select Daily Log. 2. Select Add Activity. 3. Select the building through the drop down menu. 4. Enter the Activity Type. 5. Add the Respondent. 6. Enter the Description. 7. Enter the Comments. 8. Save.
Variations	Editing Entry 1. Select Daily Log. 2. Select previous journal entry. 3. Click Edit. 4. Enter changes to entry. 5. Save.
Non-Functional	If the user attempts to access unauthorized information, the system will display a pop up window stating that the user is not authorized to access this information.
Related Misuse Cases	MC-01, MC-06, MC-07, MC-08, MC-11, MC-12, MC-13, MC-14, MC-15, MC-16, MC-17, MC-18, MC-19, MC-20, MC-21, MC-22

Number	UC- 06
Use Case	Install the Asset Management System
Description	System Administrator wants to install the Asset Management System on the network
Actors	System Administrator
Assumptions	The Sys Admin has control over the network
Steps	Steps for Pre-Determined Windows Server(s) 1. Install/confirm IIS. 2. Install/confirm Map Guide with Map Guide Author option. 3. Install/confirm database engine (Sybase or ACL, etc.). 4. Copy client database file to server (assuming that client database file was previously created and configured). 5. Configure ODBC System DSN and confirm connectivity to database. 6. Confirm that line in vbdefs.asp references the configured ODBC System DSN name. 7. Configure Web site in IIS - assign Web site name (e.g., Asset Management System) - associate with IP address assigned to server - do not allow anonymous access - specify Integrated Windows authentication - specify home directory path - specify default content page 8. Create necessary virtual directories in IIS making sure that pathing matches code references. 9. Allow access to EP document repository folder to designated High-Level user. 10. Copy files to IIS server Web site and virtual directories. 11. Register Asset Management System Web site name in local DNS server(s) using IP address (es) assigned in IIS. Steps for Developmental Workstation(s) 1. Install/confirm ARCHIBUS/FM on Asset Management System developmental workstation. 2. Create project in ARCHIBUS/FM pointing to database installed on server. 3. Confirm connectivity between ARCHIBUS and database. 4. Confirm access to ARCHIBUS database according to the security level assigned in the afm_users table. 5. Install/confirm AutoCAD and configure with ARCHIBUS Overlay on Asset Management System developmental workstation. 6. Confirm connectivity between AutoCAD and ARCHIBUS

	project.
	7. Install pre-configured SDF Loader program.
	8. Confirm connectivity to the IIS server (e.g., Ping server name)
	9. Confirm connectivity to the Asset Management System Web site (e.g., Ping Web site name).
	10. Configure Internet Explorer settings for Intrasite security and Advanced security and settings.
	11. Confirm access to the Asset Management System Web site using Internet Explorer browser.
	Steps for Asset Management System User Workstation(s)
	1. Confirm connectivity to the IIS server (e.g., Ping server name).
	2. Confirm connectivity to the Asset Management System Web site (e.g., Ping Web site name).
	3. Configure Internet Explorer settings for Intra-site security and Advanced security and settings.
	4. Confirm access to the Asset Management System Web site using Internet Explorer browser.
Variations	
Non-Functional	
Related Misuse Cases	MC-01, MC-02, MC-03, MC-04, MC-05, MC-08, MC-09, MC-10, MC-12, MC-13, MC-14, MC-15, MC-16, MC-17, MC-18, MC-19, MC-20, MC-21, MC-22

Number	UC-07
Use Case	Create Links
Description	High-Level users will have the ability to access the Asset Management System and create links to EP procedures/docs, etc.
Actors	High-Level User, or System Administrator
Assumptions	This assumes that: - System Admin has added write privileges to the ACL of the document repository folder - System is available - Data entered is correct
Steps	1. User logs into developmental workstation with assigned network username and password. 2. The system authorizes and authenticates the user and the allowed into the system. 3. User enters data into ARCHIBUS/FM tables 'ep_procedures' and 'ep_bl_doc_link' to denote document path, document name, and related building. 4. User copies documents to IIS virtual directory designated as document repository whose path agrees with that entered in the above step. 5. User confirms that the Asset Management System Web site function displays document listing and document correctly.
Variations	
Non-Functional	If the user attempts to access unauthorized information, the system will display a pop up window stating that the user is not authorized to access this information. If the user attempts to access an unauthorized network folder, the user will be notified of insufficient privileges.
Related Misuse Cases	MC-01, MC-02, MC-03, MC-04, MC-05, MC-08, MC-09, MC-10, MC-12, MC-13, MC-14, MC-15, MC-16, MC-17, MC-18, MC-19, MC-20, MC-21, MC-22

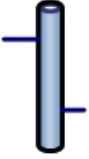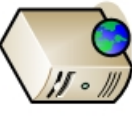| Number	UC- 08
Use Case	ARCHIBUS Administration Adding a User and Assigning Privileges
Description	The ARCHIBUS Administrator adds a user to the afm_users table so that the user will have the ability to use the Asset Management System. The user must also assign the proper privileges associated with their user-level.
Actors	ARCHIBUS Administrator
Assumptions	The ARCHIBUS Admin has the proper security privileges.
Steps	Add Individual 1. Open ARCHIBUS. 2. Select the project (in this case, it is Asset Management System but varies according to client). 3. Navigate to System Management. 4. Select Security. 5. Click the Secure Padlock. 6. Select Users. 7. Open a new record. 8. Enter the username (must match the login name). 9. Select the user-level (Review, Edit…). 10. Assign groups. Add Group 1. Open ARCHIBUS. 2. Select Security Groups. 3. Add new record. 4. Add group name. 5. Add Description.
Variations	Go directly to the data through ARCHIBUS
Non-Functional	No user password
Related Misuse Cases	MC-01, MC-02, MC-03, MC-04, MC-05, MC-08, MC-09, MC-10, MC-12, MC-13, MC-14, MC-15, MC-16, MC-17, MC-18, MC-19, MC-20, MC-21, MC-22

Appendix D   System Architecture & Use Case Diagrams

The system's architecture provides an overall structural view of the client's intact system without user involvement. The use case diagrams provide a visual aid and system interaction view of user involvement.

LEGEND

Machine Border

Local Area
Network Connection

Data Access
Connection

File Access
Connection

Control Transfer
Connection

Callee Port

Caller Port

Read Access

Write Access

Read/ Write Access

Application

Application Plug-in/
Component

Server

Database
Management System

File(s)

User

SYSTEM ARCHITECTURE

AMS User W2K Workstation

Microsoft Internet Explorer (Ver 6)

Microsoft Office 2000

Windows Server (2003 SE)

EPP Documents *.asp, *.css, *.js, *.gif, *.jpg files for website

.dwg files for AutoCAD

.sdf files for MapGuide

MapGuide Author

MapGuide Ver (6.3)

Microsoft IIS (Ver 6.0)

Sybase Central (Ver 4.1)

AMS Developmental W2K Workstation

SDF Loader Program

Development Tools

MS MapPoint

Edit Plus

MenuMaker

MS Word 2000

AutoCad 2004 w/ Archibus Overlay

Archibus-FM (Ver 14.1)
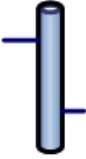
TYPICAL NETWORK TOPOLOGY FOR ASSET MANAGEMENT SYSTEM

Table of Contents

LEGEND

Machine Border

Local Area Network Connection

Data Access Connection

File Access Connection

Control Transfer Connection

Callee Port

Caller Port

Read Access

Write Access

Read/ Write Access

Application

Application Plug-in/ Component

Server

Database Management System

File(s)

User

i

UC-01: View Floor Plans

Windows Server (2003 SE)

AMS User W2K Workstation

- Microsoft Internet Explorer (Ver 6)
- Microsoft Office 2000

Step 1. Enter Username/Password
Step 2. Authenticate to AMS
Step 3. Query Floor Plans / Returned to User
Step 4. Results Returned to User

*.sdf files for MapGuide

MapGuide Author

MapGuide Ver (6.3)

*.dwg files for AutoCAD

Microsoft IIS (Ver 6.0)

EPP Documents *.asp, *.css, *.js, *.gif, *.jpg files for website

Username/Password is Verified

Sybase Central (Ver 4.1)

AMS Developmental W2K Workstation

SDF Loader Program

Development Tools

- Edit Plus
- MenuMaker
- MS MapPoint
- MS Word 2000

AutoCad 2004 w/ Archibus Overlay

Archibus-FM (Ver 14.1)

1

UC-02: Damage Assessment

AMS User W2K Workstation

Internet Explorer (Ver 6)

Microsoft Office 2000

Step 1. Enter Username/Password

User

Windows Server (2003 SE)

Step 2. Authenticate to AMS

Step 5. Enter Damage Assessment

Step 4. Displays Floor Plan

Step 3. Select Affected Area

*.sdf files for MapGuide

MapGuide Author

MapGuide Ver (6.3)

*.dwg files for AutoCAD

Microsoft IIS (Ver 6.0)

Username/Password is Verified

EPP Documents *.asp, *.css, *.js, *.gif, *.jpg files for website

Sybase Central (Ver 4.1)

AMS Developmental W2K Workstation

SDF Loader Program

Development Tools

MS MapPoint MS Word 2000 MenuMaker

Edit Plus

AutoCad 2004 w/ Archibus Overlay

Archibus-FM (Ver 14.1)

UC-03: Mark Up/ Create Floor Plans

AMS User W2K Workstation

Microsoft Internet Explorer (Ver 6)

Microsoft Office 2000

Windows Server (2003 SE)

*.sdf files for MapGuide

MapGuide Author

MapGuide Ver (6.3)

*.dwg files for AutoCAD

Microsoft IIS (Ver 6.0)

EPP Documents *.asp, *.css, *.js, *.gif, *.jpg files for website

Sybase Central (Ver 4.1)

AMS Developmental W2K Workstation

SDF Loader Program

Step 4. Define Base Drawing

Step 1. Create Floor Plan Drawing Using AutoCad

Steps 2 and 3. Add and Link Assets to Drawing Using Archibus

Development Tools

MenuMaker

MS MapPoint MS Word 2000

AutoCad 2004 w/ Archibus Overlay

Archibus-FM (Ver 14.1)

User

3

UC-04: Find Specialized Employees

Windows Server (2003 SE)

AMS User W2K Workstation

Microsoft Internet Explorer (Ver 6)

Microsoft Office 2000

Step 1. Enter Username/Password

User

*.sdf files for MapGuide

*.dwg files for AutoCAD

EPP Documents *.asp, *.css, *.js, *.gif, *.jpg files for website

Step 2. Authenticate to AMS

Step 3. Enter Personnel Parameters

Step 4. Personnel are Retrieved

MapGuide Ver (6.3)

Microsoft IIS

Step 5. Relevant Personnel ...

Username/Password is Verified

Sybase Central (Ver 4.1)

AMS Developmental W2K Workstation

SDF Loader Program

Development Tools

Edit Plus

MenuMaker

MS MapPoint MS Word 2000

AutoCad 2004 w/ Archibus Overlay

Archibus-FM (Ver 14.1)

4

UC-05: Journal Entry

Windows Server (2003 SE)

EPP Documents *.asp, *.css, *.js, *.gif, *.jpg files for website

*.dwg files for AutoCAD

*.sdf files for MapGuide

Username/Password is Verified

MapGuide Ver (6.3)

Microsoft IIS (Ver 6.0)

Sybase Central (Ver 4.1)

AMS Developmental W2K Workstation

Development Tools

Edit Plus

MenuMaker

MS MapPoint MS Word 2000

SDF Loader Program

AutoCad 2004 w/ Archibus Overlay

Archibus-FM (Ver 14.1)

AMS User W2K Workstation

Microsoft Internet Explorer (Ver 6)

Microsoft Office 2000

Step 1. Enter Username/Password

User

Step 2. Authenticate to AMS

Step 3. Navigate to Facility Homepage

Enter Activity Type and Comments

Step 4. Enter Activity Type and Comments

UC-06: Install the Asset Management System

Windows Server (2003 SE)

Steps for AMS User Workstation

Step 23: Confirm connectivity to the IIS server

Step 24: Confirm connectivity to the AMS website

Step 25: Configure Internet Explorer settings for Intrasite security and Advanced security and settings.

Step 26: Confirm access to the AMS website using Internet Explorer browser

AMS User W2K Workstation

Microsoft Internet Explorer (Ver 6)

Microsoft Office 2000
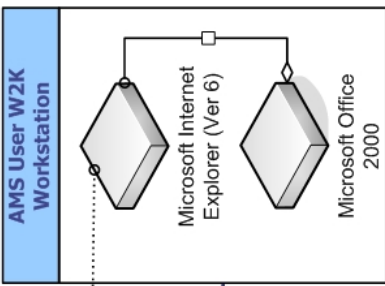
Steps for Windows Server 2003

Step 4: Copy Client Database file to Server

Step 5: Configure ODBC System DSN and confirm connectivity to database.

Step 6: Confirm that line in vbdefs.asp references the configured ODBC System DSN name.

Step 11: Register AMS website name in local DNS server(s) using IP address (es) assigned in IIS.

Steps for Developmental Workstation

Step 15: Confirm access to Archibus database according to the security level assigned in the afm_users table.

Step 20: Confirm connectivity to the Asset Management System website

Step 21: Configure Internet Explorer settings for Intrasite security and Advanced security and settings.

Step 22: Confirm access to the AMS website using Internet Explorer browser.

Installer

*.sdf files for MapGuide

MapGuide Author

MapGuide (Ver 6.3)

Step 2: Install/confirm Map Guide with Map Guide Author option.

Step 10: Copy files to IIS server website and virtual directories

*.dwg files for AutoCAD

Step 1: Install/Confirm IIS

Microsoft IIS (Ver 6.0)

*.jpg's, *.gif, *.ipp's repository folder

EPP Documents

Step 9: Allow access to *.jpg's, *.gif, *.ipp document

Step 7: Configure IIS

Step 8: Create necessary virtual directories in IIS

Sybase Central (Ver 4.1)

AMS Developmental W2K Workstation

SDF Loader Program

Step 19: Confirm connectivity to the IIS server

Development Tools

Edit Plus

MenuMaker

MS MapPoint

MS Word 2000

Step 16: Install/confirm AutoCAD and configure AutoCAD and Archibus Overlay

AutoCad 2004 w/ Archibus Overlay

Step 17: Confirm AutoCAD and Archibus-FM

Step 18: Confirm connectivity between AutoCAD and Archibus project.

Step 12: Install/confirm Archibus-FM

Archibus-FM (Ver 14.1)

Step 13: Create project in Archibus-FM pointing to DB on server.

Step 14: Confirm connectivity between Archibus and database

6

UC-07: Create Links

Windows Server (2003 SE)

AMS User W2K Workstation

Microsoft Internet Explorer (Ver 6)

Microsoft Office 2000

Step 3. Confirm AMS Website Function

User

*.sdf files for MapGuide

MapGuide Author

MapGuide Ver (6.3)

*.dwg files for AutoCAD

Microsoft IIS (Ver 6.0)

EPP Documents *.asp, *.css, *.js, *.gif, *.jpg files for website

Step 2. Copy

Document Directory

Virtual Directory

Sybase Central (Ver 4.1)

Step 1. Enter Data into Archibus-FM tables 'ep_procedures' and 'ep_bl_doc_link'

AMS Developmental W2K Workstation

SDF Loader Program

Development Tools

MS MapPoint

Edit Plus

MenuMaker

MS Word 2000

AutoCad 2004 w/ Archibus Overlay

Archibus-FM (Ver 14.1)

7

UC-08: Adding a User and Assigning Privileges

UC-09: Adding a Group

AMS User W2K Workstation

Microsoft Internet Explorer (Ver 6)

Microsoft Office 2000

Windows Server (2003 SE)

*.sdf files for MapGuide

MapGuide Author

MapGuide Ver (6.3)

*.dwg files for AutoCAD

EPP Documents *.asp, *.css, *.js, *.gif, *.jpg files for website

Microsoft IIS (Ver 6.0)

Sybase Central (Ver 4.1)

AMS Developmental W2K Workstation

SDF Loader Program

Development Tools

Edit Plus

Step 1. Select Security Groups

MenuMaker

MS MapPoint MS Word 2000

AutoCad 2004 w/ Archibus Overlay

Archibus-FM (Ver 14.1)

Step 4. Add Description

Step 3. Add a Group Name

Step 2. Add a New Record

User

Appendix E   Misuse Cases

The misuse cases that follow outline all possible threats, vulnerabilities, and misuses that may affect the components in the Asset Management System. They also provide architectural and policy recommendations for security experts to prevent, detect, and recover from system misuses and cyber-attacks [Meier 03].

Number:	MC-01
Name:	Unauthorized logon on the Windows 2003 server
Scope:	User Authorization Concerns
Priority:	___ Low ___ Medium _x_ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Unauthorized users
Access Right Levels:	_x_ Low-Level System Users _x_ Medium-Level System Users _x_ High-Level System Users _x_ Sys Admin _x_ Other Network User
Point of Entry:	___ Network _x_ Host ___ Application
Security Attributes Affected:	_x_ Confidentiality _x_ Integrity ___ Availability
Description:	An unauthorized user attempts to log on to the Windows 2003 server and succeeds.
Sophistication:	_x_ Low ___ Medium ___ High
Pre-conditions:	• ACLs are configured properly in a domain based network. • The unauthorized user has unintended logon rights to the Windows 2003 server. • The Windows 2003 server resides on an intranet network
Assumptions:	• The user does not have permission to log on to the Windows 2003 server.

Post-conditions:	Worst Case Threat:	• The unauthorized user logs onto the Windows 2003 server machine. His/her actions are never caught.
	Wanted Prevention Guarantee:	• Enforce machine ACL security policy. (role-based user authentication)
	Wanted Detection Guarantee:	• Logon attempts are logged and viewed by system administrators.
	Wanted Recovery Guarantee:	• Remove users' unauthorized logon rights on the server.

Potential Mis-actor Profiles:	Medium to highly skilled, potentially host administrators with medium criminal intent
Stakeholders and Threats:	• AMS Client Company: loss of data integrity and/or confidentiality • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-01, UC-02, UC-03, UC-04, UC-05, UC-06, UC-07, UC-08

Related Threats:	Elevation of Privilege, Disclosure of Confidential Data, Unauthorized Access to Administration Interface, Unauthorized Access to Configuration Stores, Retrieval of Print Text Configuration Secrets
Architectural Recommendation:	• (AR-01) All shared drives on the network should enforce authentication policies. • (AR-03) Audit information is stored in a separate location from the servers and the workstations. • (AR-19) Implement role-based authentication control.
Policy Recommendation:	• (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-04) Applications and operating systems must be patched routinely. (Bi-Monthly) • (PR-07) Enforce strong password policies. • (PR-13) Password protect any necessary shared documents. • (PR-16) Require users to change their passwords periodically. (Monthly) • (PR-19) Set clear and defined user access controls for all users. (Low, Medium, High, System Admins) • (PR-20) Perform routine system and data backup. (Weekly) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly) • (PR-23) Users should not have rights or access levels beyond those prescribed by their job responsibilities. • (PR-24) Users should not reveal their account names and passwords in any situation.

Number:	MC-02
Name:	Sys admin gains access to system data
Scope:	User Authorization Concerns
Priority:	___ Low _x_ Medium ___ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Sys Admin
Access Right Levels:	___ Low-Level System User ___ Medium-Level System User ___ High-Level System User _x_ Sys Admin ___ Other Network User
Point of Entry	___ Network ___ Host _x_ Application
Security Attributes Affected:	_x_ Confidentiality _x_ Integrity ___ Availability
Description:	A sys admin attempts to gain access data on the Windows 2003 server and succeeds.
Sophistication:	___ Low _x_ Medium ___ High
Pre-conditions:	• The sys admin has logon rights to the Windows 2003 server or he/she has the credentials to access the database.
Assumptions:	• The sys admin does not have permission to access data on the Windows 2003 server.

Post-conditions:	Worst Case Threat:	• The sys admin sees and/or tampers with the system data. His/her actions are never caught.
	Wanted Prevention Guarantee:	• Enforce machine ACL security policy. Separate credentials for system administration and application access. (role-based user authentication)
	Wanted Detection Guarantee:	• Logon attempts are logged, application usage is logged, and database accesses are logged. Audit information is cross reviewed by a group of sys admins and managers.
	Wanted Recovery Guarantee:	• Restore data from backup if data is tampered with.

Potential Mis-actor Profiles:	Highly skilled system administrators who understand how the system works and know about backdoors (if any exist).
Stakeholders and Threats:	• AMS Client Company: loss of data confidentiality and/or integrity • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-06, UC-07, UC-08

Related Threats:	Disclosure of Confidential Data, Access to Sensitive Data Storages
Architectural Recommendation:	• (AR-03) Audit information is stored in a separate location from the servers and the workstations. • (AR-19) Implement role-based authentication control.
Policy Recommendation:	• (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-04) Configuration changes are stored and cross-reviewed. (Monthly) • (PR-15) Perform routine system and data backup. (Weekly) • (PR-18) Separate personnel review of sys admin's activities. (Monthly) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly) • (PR-23) Users should not have rights or access levels beyond those prescribed by their job responsibilities.

Number:	MC-03
Name:	Users gain sys admin rights on the Windows 2003 server (Elevation of Privilege)
Scope:	User Authorization Concerns
Priority:	__ Low __ Medium _x_ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Users
Access Right Levels:	___ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User ___ Sys Admin-Level System User _x_ Other Network User
Point of Entry:	___ Network _x_ Host ___ Application
Security Attributes affected:	_x_ Confidentiality _x_ Integrity ___ Availability
Description:	A user attempts to gain sys admin rights on the Windows 2003 server and succeeds.
Sophistication:	___ Low ___ Medium _x_ High
Pre-conditions:	• The user has unintended logon rights to the Windows 2003 server.
Assumptions:	• The user is not already a sys admin. • The user does not have expressed permission to gain sys admin rights.

Post-conditions:	Worst Case Threat:	• The user gains sys admin rights on the Windows 2003 server and then tampers with system and/or user data. His/her actions are never caught.
	Wanted Prevention Guarantee:	• Enforce machine ACL security policy. (role-based user authentication)
	Wanted Detection Guarantee:	• Logon attempts are logged and viewed by system administrators.
	Wanted Recovery Guarantee:	• Remove users' unauthorized logon rights on the server.

Potential Mis-actor Profiles:	Highly skilled users with high criminal intent.
Stakeholders and Threats:	• AMS Client Company: loss of data integrity and/or confidentiality • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-06, UC-07, UC-08
Related Threats:	Elevation of Privileges, Unauthorized Access to Administration Interfaces, Unauthorized Access to Configuration Stores

Architectural Recommendation:	• (AR-03) Audit information is stored in a separate location from the servers and the workstations. • (AR-19) Implement role-based authentication control.
Policy Recommendation:	• (PR-02) Applications and operating systems must be patched routinely. (Bi-Monthly) • (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-04) Configuration changes are stored and cross-reviewed. (Monthly) • (PR-07) Enforce strong password policies. • (PR-13) Password protect any necessary shared documents. • (PR-16) Require users to change their passwords periodically. (Monthly) • (PR-20) Perform routine system and data backup. (Weekly) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly) • (PR-22) Users should log out of AMS system or close browser as soon as their activities are done. • (PR-23) Users should not have rights or access levels beyond those prescribed by their job responsibilities. • (PR-24) Users should not reveal their account names and passwords in any situation.

Number:	MC-04
Name:	Sys admin deletes critical system configurations on the Windows 2003 server.
Scope:	System integrity concerns
Priority	___ Low ___ Medium _x_ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Sys Admin
Access Right Levels:	___ Low-Level System User ___ Medium-Level System User ___ High-Level System User _x_ Sys Admin ___ Other Network User
Point of Entry	___ Network _x_ Host ___ Application
Security Attributes affected:	___ Confidentiality _x_ Integrity _x_ Availability
Description:	A malicious sys admin attempts to delete critical system configurations on the Windows 2003 server without authorization and succeeds. Examples include deleting user accounts, removing user access rights, uninstalling system applications.
Sophistication:	_x_ Low ___ Medium ___ High
Pre-conditions:	• The sys admin has authorized access rights to the Windows 2003 server.
Assumptions:	• The sys admin does not have permission to delete critical system configuration on the Windows 2003 server.

Post-conditions:	Worst Case Threat:	• Critical system configurations are lost and irrecoverable.
	Wanted Prevention Guarantee:	• System configurations are backed up every day and stored offsite.
	Wanted Detection Guarantee:	• Audit information will be cross-reviewed by a group of sys admins and managers.
	Wanted Recovery Guarantee:	• System configurations are restored from previous backups.

Potential Mis-actor Profiles:	Highly skilled system administrators with high criminal intent.
Stakeholders and Threats:	• AMS Client Company: loss of data integrity and/or confidentiality • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-06, UC-07, UC-08

Related Threats:	Unauthorized Access to Configuration Stores
Architectural Recommendation:	• (AR-03) Audit information is stored in a separate location from the servers and the workstations.
Policy Recommendation:	• (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-04) Configuration changes are stored and cross-reviewed. (Monthly) • (PR-18) Separate personnel review of sys admin's activities. (Monthly) • (PR-20) Perform routine system and data backup. (Weekly) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly)

Number:	MC-05
Name:	Sys admin creates holes in the system configurations on the Windows 2003 server.
Scope:	System integrity concerns
Priority:	___ Low _x_ Medium ___ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Sys Admin
Access Right Levels:	___ Low-Level System User ___ Medium-Level System User ___ High-Level System User _x_ Sys Admin ___ Other Network User
Point of Entry:	___ Network _x_ Host ___ Application
Security Attributes affected:	_x_ Confidentiality _x_ Integrity _x_ Availability
Description:	A sys admin creates holes and/or backdoors in the system configurations on the Windows 2003 server. Examples include installing remote control applications, setting up secret admin accounts, using blank passwords
Sophistication:	_x_ Low ___ Medium ___ High
Pre-conditions:	• The sys admin has authorized access rights to the Windows 2003 server.
Assumptions:	• The sys admin does not have permission to modify system configurations on the Windows 2003 server. • Holes/backdoors create vulnerabilities for the AMS system.

Post-conditions:	Worst Case Threat:	• System is susceptible to future attacks.
	Wanted Prevention Guarantee:	• System configurations are backed up every day and stored offsite. Vulnerability detection systems are installed and used to monitor the AMS system.
	Wanted Detection Guarantee:	• Logon attempts are logged and viewed by a group of sys admins and managers.
	Wanted Recovery Guarantee:	• System configurations are restored from previous backups.

Potential Mis-actor Profiles:	Highly skilled system administrators who may or may not have criminal intent.
Stakeholders and Threats:	• AMS Client Company: breach of system integrity • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-06, UC-07, UC-08

Related Threats:	Lack of individual accountability, Bypassing Auditing and Logging
Architectural Recommendation:	• (AR-03) Audit information is stored in a separate location from the servers and the workstations.
Policy Recommendation:	• (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-04) Configuration changes are stored and cross-reviewed. (Monthly) • (PR-18) Separate personnel review of sys admin's activities. (Monthly) • (PR-20) Perform routine system and data backup. (Weekly) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly)

Number:	MC-06
Name:	User deletes critical data from the AMS system.
Scope:	Data Integrity concerns
Priority:	___ Low ___ Medium _x_ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Users
Access Right Levels:	___ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User ___ Sys Admin ___ Other Network User
Point of Entry	___ Network ___ Host _x_ Application
Security Attributes affected:	___ Confidentiality _x_ Integrity _x_ Availability
Description:	A malicious user attempts to delete critical data from the AMS system without authorization and succeeds.
Sophistication:	_x_ Low ___ Medium ___ High
Pre-conditions:	• The user has rights to modify AMS system data.
Assumptions:	• The user does not have permission to delete critical system configurations on the Windows 2003 server.

Post-conditions:	Worst Case Threat:	• Critical system data are lost and irrecoverable.
	Wanted Prevention Guarantee:	• System data are backed up every day and stored offsite.
	Wanted Detection Guarantee:	• System logon attempts and application accesses are logged and viewed by a group of sys admins and managers. • Enforce role-based user authentication
	Wanted Recovery Guarantee:	• System data are restored from previous backups.

Potential Mis-actor Profiles:	Users with high criminal intent.
Stakeholders and Threats:	• AMS Client Company: loss of data integrity and/or availability • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-02, UC-03, UC-05
Related Threats:	Data Tampering

Architectural Recommendation:	• (AR-03) Audit information is stored in a separate location from the servers and the workstations. • (AR-19) Implement role-based authentication control.
Policy Recommendation:	• (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-04) Configuration changes are stored and cross-reviewed. (Monthly) • (PR-13) Password protect any necessary shared documents. • (PR-14) Perform information integrity checks on a routine basis, reviewed by DBA and/or managers. (Bi-Weekly) • (PR-20) Perform routine system and data backup. (Weekly) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly) • (PR-22) Users should log out of AMS system or close browser as soon as their activities are done. • (PR-23) Users should not have rights or access levels beyond those prescribed by their job responsibilities.

Number:	MC-07	
Name:	Users falsify system data	
Scope:	Data integrity concerns	
Priority:	___ Low _x_ Medium ___ High	
Deployment Environment:	_x_ Intranet ___ Extranet/Internet	
Mis-actors:	Users	
Access Right Levels:	_x_ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User ___ Sys Admin ___ Other Network User	
Point of Entry:	___ Network ___ Host _x_ Application	
Security Attributes affected:	___ Confidentiality _x_ Integrity ___ Availability	
Description:	A user enters false data into the system.	
Sophistication:	_x_ Low ___ Medium ___ High	
Pre-conditions:	• The user has sufficient rights to the system data.	
Assumptions:	• The user is logged in.	
Post-conditions:	Worst Case Threat:	• False data hinders the decision-making process.
	Wanted Prevention Guarantee:	• System data are backed up every day and stored offsite properly. • Implement role-based authentication system.
	Wanted Detection Guarantee:	• Another person reviews changes to the system data. • Develop scripts to perform integrity checking between current database state and backup database state. • Enforce role-based user authentication
	Wanted Recovery Guarantee:	• System data can be restored from the backup.
Potential Mis-actor Profiles:	Medium to highly skilled, potentially host administrators with medium criminal intent.	
Stakeholders and Threats:	• AMS Client Company: loss of data integrity and/or confidentiality • Acme Group: loss of reputation, loss of current and potential clients	
Related Use Cases:	UC-02, UC-03, UC-05	

Related Threats:	Data Tampering
Architectural Recommendation:	• (AR-03) Audit information is stored in a separate location from the servers and the workstations. • (AR-19) Implement role-based authentication control.
Policy Recommendation:	• (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-14) Perform information integrity checks on a routine basis, reviewed by DBA and/or managers. (Bi-Weekly) • (PR-20) Perform routine system and data backup. (Weekly) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly) • (PR-22) Users should log out of AMS system or close browser as soon as their activities are done.

Number:	MC-08
Name:	Access system data through developmental machines
Scope:	Data integrity concerns
Priority:	___ Low ___ Medium _x_ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Users, sys admins
Access Right Levels:	_x_ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User _x_ Sys Admin _x_ Other Network User
Point of Entry:	_x_ Network _x_ Host ___ Application
Security Attributes affected:	_x_ Confidentiality _x_ Integrity ___ Availability
Description:	A user or a sys admin accesses system data through developmental machines.
Sophistication:	___ Low _x_ Medium ___ High
Pre-conditions:	• The mis-actor has access to the developmental machine.
Assumptions:	• The user is logged in.

Post-conditions:	Worst Case Threat:	• Critical data is deleted or modified, false data is added. Original data is lost and irrecoverable.
	Wanted Prevention Guarantee:	• System data are backed up every day and stored offsite properly. • Implement role-based user authentication.
	Wanted Detection Guarantee:	• Another person reviews changes to the system data. Access to developmental machines are logged and controlled.
	Wanted Recovery Guarantee:	• System data can be restored from the backup.

Potential Mis-actor Profiles:	Medium or highly skilled users, high-skilled sys admin, may or may not have criminal intent
Stakeholders and Threats:	• AMS Client Company: loss of data integrity and confidentiality • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-01, UC-02, UC-03, UC-04, UC-05, UC-06, UC-07, UC-08

Related Threats:	Disclosure of Confidential Data, Data Tampering, Luring Attack, Unauthorized Access to Configuration Stores, Lack of Individual Accountability, Over-Privileged Process and Service Accounts
Architectural Recommendation:	• (AR-08) Developmental machines should have strong access control mechanisms. • (AR-19) Implement role-based authentication control.
Policy Recommendation:	• (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-06) Do not set up shared files/folders/drives on the AMS network server or workstation. • (PR-07) Enforce strong password policies. • (PR-09) Follow the principle of least privilege[3] and use least privileged service accounts to run processes and access resources. • (PR-10) Log all incoming and outgoing traffic (IIS, database engine, MapGuide, firewall). • (PR-13) Password protect any necessary shared documents. • (PR-14) Perform information integrity checks on a routine basis, reviewed by DBA and/or managers. (Bi-Weekly) • (PR-19) Set clear and defined user access controls for all users. (Low, Medium, High, System Admins). • (PR-20) Perform routine system and data backup. (Weekly) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly) • (PR-23) Users should not have rights or access levels beyond those prescribed by their job responsibilities.

---

[3]   The security guideline that a user should have the minimum privileges necessary to perform a specific task. This helps to ensure that, if a user is compromised, the impact is minimized by the limited privileges held by that user. In practice, a user runs within the security context of a normal user. When a task requires additional privileges, the user can use a tool such as **Run as** to start a specific process with those additional privileges or to log on as a user with the necessary privileges.

Number:	MC-09
Name:	Access system data directly to/from database
Scope:	Data integrity concerns
Priority:	___ Low _x_ Medium ___ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	System administrators, developers
Access Right Levels:	___ Low-Level System User ___ Medium-Level System User _x_ High-Level System User _x_ Sys Admin _x_ Other Network User
Point of Entry:	___ Network ___ Host _x_ Application
Security Attributes affected:	_x_ Confidentiality _x_ Integrity ___ Availability
Description:	A system administrator or high-level user accesses system data directly to/from databases.
Sophistication:	___ Low _x_ Medium ___ High
Pre-conditions:	• The mis-actor has access to the database server. • The mis-actor has database credentials.
Assumptions:	

Post-conditions:	Worst Case Threat:	• Critical data is deleted or modified, false data is added. Original data is lost and irrecoverable.
	Wanted Prevention Guarantee:	• System data are backed up every day and stored offsite properly. Control database access credentials.
	Wanted Detection Guarantee:	• Log database access. Develop scripts to perform integrity checking between current database state and backup database state.
	Wanted Recovery Guarantee:	• Monitor and log database access. Perform integrity checking. If errors found, backup database will be restored.

Potential Mis-actor Profiles:	Medium to highly skilled administrators or developers who may or may not have criminal intent.
Stakeholders and threats:	• AMS Client Company: loss of data integrity and confidentiality • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-06, UC-07, UC-08

Related Threats:	Disclosure of Confidential Data, Data Tampering, Luring Attack, Unauthorized Access to Configuration Stores, Lack of Individual Accountability, Over-Privileged Process and Service Accounts
Architectural Recommendation:	• (AR-07) Database activities should be logged and stored in a separate secure server. • (AR-19) Implement role-based authentication control.
Policy Recommendation:	• (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-04) Configuration changes are stored and cross-reviewed. (Monthly) • (PR-09) Follow the principle of least privilege and use least privileged service accounts to run processes and access resources. • (PR-13) Password protect any necessary shared documents • (PR-14) Perform information integrity checks on a routine basis, reviewed by DBA and/or managers. (Bi-Weekly) • (PR-20) Perform routine system and data backup. (Weekly) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly) • (PR-23) Users should not have rights or access levels beyond those prescribed by their job responsibilities.

Number:	MC-10
Name:	Steal user credential information through developmental machines
Scope:	Confidentiality concerns
Priority:	___ Low ___ Medium _x_ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Malicious high-level users or sys admins
Access Right Levels:	___ Low-Level System User ___ Medium-Level System User _x_ High-Level System User _x_ Sys Admin _x_ Other Network User
Point of Entry:	___ Network ___ Host _x_ Application
Security Attributes affected:	_x_ Confidentiality _x_ Integrity ___ Availability
Description:	A malicious high-level user or a sys admin accesses the database from the developmental machines and retrieves user credential information.
Sophistication:	___ Low ___ Medium _x_ High
Pre-conditions:	• The mis-actor understands how the system works and where user credentials are stored.
Assumptions:	• The mis-actor should not have access to developmental machines.

Post-conditions:	Worst Case Threat:	• User credentials are stolen and misused.
	Wanted Prevention Guarantee:	• Access from developmental machines to database is logged and controlled. • Enforce policy for every user to change password periodically. • Passwords should be encrypted in the database. • Implement authentication mechanisms (dongles, smart cards, etc.) to authenticate proper users in the developmental machines. • Implement role-based user authentication.
	Wanted Detection Guarantee:	• Monitor unusual login patterns.
	Wanted Recovery Guarantee:	• Any access from developmental machines to database is logged and controlled.

Potential Mis-actor Profiles:	Highly skilled users or sys admins with high criminal intent.
Stakeholders and Threats:	• AMS Client Company: loss of data confidentiality • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-06, UC-07, UC-08
Related Threats:	Elevation of Privilege, Disclosure of Confidential Data, Unauthorized Access to Administration Interfaces, Unauthorized Access to Configuration Stores, Retrieval of Plaintext Configuration Secretes, Over-Privileged Process and Service Accounts, Credential Theft
Architectural Recommendation:	• (AR-12) Encrypt user credentials in configurations and databases. • (AR-19) Implement role-based authentication control.
Policy Recommendation:	• (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-06) Do not set up shared files/folders/drives on the AMS network server or workstation. • (PR-07) Enforce strong password policies. • (PR-13) Password protect any necessary shared documents. • (PR-14) Perform information integrity checks on a routine basis, reviewed by DBA and/or managers. (Bi-Weekly) • (PR-16) Require users to change their passwords periodically. (Monthly) • (PR-19) Set clear and defined user access controls for all users. (Low, Medium, High, System Admins). • (PR-20) Perform routine system and data backup. (Weekly) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly) • (PR-23) Users should not have rights or access levels beyond those prescribed by their job responsibilities. • (PR-24) Users should not reveal their account names and passwords in any situation.

Number:	MC-11	
Name:	A user sees data that he or she should not see from his or her workstation.	
Scope:	Data confidentiality concerns	
Priority:	___ Low _x_ Medium ___ High	
Deployment Environment:	_x_ Intranet ___ Extranet/Internet	
Mis-actors:	Users	
Access Right Levels:	_x_ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User ___ Sys Admin _x_ Other Network User	
Point of Entry:	___ Network _x_ Host ___ Application	
Security Attributes affected:	_x_ Confidentiality ___ Integrity ___ Availability	
Description:	A user sees data that he or she should not see from his or her browser on his or her workstation.	
Sophistication:	_x_ Low ___ Medium ___ High	
Pre-conditions:	• The user can access the system.	
Assumptions:	• The data is not approved for user's access level.	
Post-conditions:	Worst Case Threat:	• Sensitive information is viewed by the user, who doesn't have the rights to see it.
	Wanted Prevention Guarantee:	• Categorized and hierarchical access level. Data encryption is desired. • Enforce role-based user authentication.
	Wanted Detection Guarantee:	• Access to data is controlled by security and access level. • Review user activities through system and network logs.
	Wanted Recovery Guarantee:	N/A
Potential Mis-actor Profiles:	General users, may or may not have critical intent	
Stakeholders and Threats:	• AMS Client Company: loss of data confidentiality • Acme Group: loss of reputation, loss of current and potential clients	
Related Use Cases:	UC-01, UC-02, UC-03, UC-04, UC-05	

Related Threats:	Disclosure of Confidential Data
Architectural Recommendation:	• (AR-18) Implement hierarchical authorization levels. • (AR-19) Implement role-based authentication control.
Policy Recommendation:	• (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-06) Do not set up shared files/folders/drives on the AMS network server or workstation. • (PR-07) Enforce strong password policies. • (PR-09) Follow the principle of least privilege and use least privileged service accounts to run processes and access resources. • (PR-13) Password protect any necessary shared documents. • (PR-19) Set clear and defined user access controls for all users. (Low, Medium, High, System Admins). • (PR-21) User activities must be periodically reviewed. (Bi-Monthly) • (PR-23) Users should not have rights or access levels beyond those prescribed by their job responsibilities.

Number:	MC-12
Name:	Malicious user uses replay attack in the same browser to assume the identity of another user.
Scope:	Confidentiality concerns
Priority:	___ Low _x_ Medium ___ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Users, sys admins
Access Right Levels:	_x_ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User _x_ Sys Admin _x_ Other Network User
Point of Entry:	___ Network ___ Host _x_ Application
Security Attributes affected:	_x_ Confidentiality _x_ Integrity ___ Availability
Description:	A malicious user uses replay attack in the same browser (e.g., pressing the back or forward button) to assume the identity of another user who previously logged into the system.
Sophistication:	___ Low _x_ Medium ___ High
Pre-conditions:	• The victim user previously logged into the system and his or her session is still live. • The malicious user is on the same machine as the victim user.
Assumptions:	• AMS system supports the concept of session. A session is kept alive for a predefined time interval.

Post-conditions:	Worst Case Threat:	• The malicious user assumes the identity of the victim user and carries out malicious actions. His/her actions are never caught.
	Wanted Prevention Guarantee:	• Shorten the timeout for session kept-alive. Use dynamic content and force pages to not save cache locally. Use HTTPS.
	Wanted Detection Guarantee:	• Monitor system and network resource usage for unusual activities.
	Wanted Recovery Guarantee:	N/A

Potential Mis-actor Profiles:	All levels of users or sys admins with high criminal intent.

Stakeholders and Threats:	• AMS Client Company: loss of data confidentiality • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-01, UC-02, UC-03, UC-04, UC-05, UC-06, UC-07, UC-08
Related Threats:	Cookie Replay Attack, Windows Integrated Authentication Replay, Credential Theft
Architectural Recommendation:	• (AR-11) Use dynamic content and force pages to not save cache locally. • (AR-32) Use HTTPS for server-to-client Web data transfer encryption. • (AR-28) Setup IIS to prompt for user credentials every time. • (AR-29) Shorten the timeout for session kept-alive.
Policy Recommendation:	• (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly) • (PR-22) Users should log out of AMS system or close browser as soon as their activities are done.

Number:	MC-13
Name:	Malicious users tap communications channel between workstations and servers
Scope:	Confidentiality concerns
Priority:	___ Low ___ Medium _x_ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Users, sys admins
Access Right Levels:	_x_ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User _x_ Sys Admin _x_ Other Network User
Point of Entry:	_x_ Network ___ Host ___ Application
Security Attributes affected:	_x_ Confidentiality ___ Integrity ___ Availability
Description:	A malicious user taps communication channels between workstations and servers. Examples include using a software sniffer programs or a hardware sniffer device.
Sophistication:	___ Low ___ Medium _x_ High
Pre-conditions:	• The victim user is on his or her workstation and is not aware of the tapping activities. • The malicious user is on the same network as the victim user.
Assumptions:	• The data transfer channel is unencrypted.

Post-conditions:	Worst Case Threat:	The malicious user steals information transferred between the server and the workstations. His/her actions are never caught.
	Wanted Prevention Guarantee:	Use encrypted data transfer such as HTTPS.
	Wanted Detection Guarantee:	Monitor system and network resource usage for sniffing activities.
	Wanted Recovery Guarantee:	N/A

Potential Mis-actor Profiles:	Highly skilled users or sys admins with high criminal intent.
Stakeholders and Threats:	• AMS Client Company: loss of data confidentiality • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-01, UC-02, UC-03, UC-04, UC-05, UC-06, UC-07, UC-08

Related Threats:	Information Gathering, Sniffing, Spoofing, Session Hijacking, Network Eavesdropping, Data Tampering, Man in Middle
Architectural Recommendation:	• (AR-25) Secure communication channels between servers and servers. • (AR-32) Use HTTPS for server-to-client Web data transfer encryption.
Policy Recommendation:	• (PR-01) All installations must be approved and reviewed by managers. • (PR-02) Applications and operating systems must be patched routinely. (Bi-Monthly) • (PR-08) Firewalls and IDS must be patched routinely. (Monthly) • (PR-10) Log all incoming and outgoing traffic (IIS, database engine, MapGuide, firewall). • (PR-12) Only sys admins are permitted to install any software and/or hardware. • (PR-13) Password protect any necessary shared documents • (PR-21) User activities must be periodically reviewed. (Bi-Monthly)

Number:	MC-14	
Name:	Malicious users gain access to sensitive data via saved Excel export files on victim's machine.	
Scope:	Data confidentiality concerns	
Priority:	___ Low _x_ Medium ___ High	
Deployment Environment:	_x_ Intranet ___ Extranet/Internet	
Mis-actors:	Users, system administrators, developers	
Access Right Levels:	_x_ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User _x_ Sys Admin-Level System User _x_ Other Network User	
Point of Entry:	___ Network _x_ Host ___ Application	
Security Attributes affected:	_x_ Confidentiality ___ Integrity ___ Availability	
Description:	A malicious user gains access to the file system of the victim user and gains access to sensitive data via the saved Excel exported data files.	
Sophistication:	_x_ Low ___ Medium ___ High	
Pre-conditions:	• The victim user left his workstation's file system open to access from others. • The malicious user is on the same network as the victim user.	
Assumptions:	• The victim saves the Excel file on his/her workstation	
Post-conditions:	Worst Case Threat:	• The malicious user steals information in the exported Excel files. His/her actions are never caught.
	Wanted Prevention Guarantee:	• Enforce security policies for file system access • Implement role-based user authentication.
	Wanted Detection Guarantee:	• Monitor system and network resource usage for unusual data transfer.
	Wanted Recovery Guarantee:	N/A
Potential Mis-actor Profiles:	All level users with high criminal intent.	
Stakeholders and Threats:	• AMS Client Company: loss of data confidentiality • Acme Group: loss of reputation, loss of current and potential clients	
Related Use Cases:	UC-01, UC-02, UC-03, UC-04, UC-05, UC-06, UC-07, UC-08	

Related Threats:	Disclosure of Confidential Data
Architectural Recommendation:	• (AR-19) Implement role-based authentication control.
Policy Recommendation:	• (PR-06) Do not set up shared files/folders/drives on the AMS network server or workstation. • (PR-13) Password protect any necessary shared documents. • (PR-14) Perform information integrity checks on a routine basis, reviewed by DBA and/or managers. (Bi-Weekly) • (PR-20) Perform routine system and data backup. (Weekly)

Number:	MC-15
Name:	Malicious users install malicious programs that can tap into Excel's memory to steal exported data.
Scope:	Data confidentiality concerns
Priority:	___ Low _x_ Medium ___ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Users, sys admins
Access Right Levels:	_x_ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User _x_ Sys Admin _x_ Other Network User
Point of Entry:	___ Network _x_ Host ___ Application
Security Attributes affected:	_x_ Confidentiality ___ Integrity ___ Availability
Description:	Malicious users install malicious programs that can tap into Excel's memory to steal exported data. Because Excel and Microsoft Office overall use shared memory, the shared memory can be tapped by programs or malicious scripts.
Sophistication:	___ Low ___ Medium _x_ High
Pre-conditions:	The malicious user gained access to the victim's machine at some point and installed malicious programs.
Assumptions:	

Post-conditions:	Worst Case Threat:	• The malicious user steals information in the exported Excel memory. His/her actions are never caught.
	Wanted Prevention Guarantee:	• Enforce security policies for workstation system access
	Wanted Detection Guarantee:	• Monitor system resource usage for unusual programs.
	Wanted Recovery Guarantee:	• Limit user privileges regarding installation of any programs.

Potential mis-actor profiles:	Highly skilled users with high criminal intent.
Stakeholders and threats:	• AMS Client Company: loss of data confidentiality • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-01, UC-02, UC-03, UC-04, UC-05, UC-06, UC-07, UC-08

Related Threats:	Disclosure of Confidential Data
Architectural Recommendation:	
Policy Recommendation:	• (PR-01) All installation must be approved and reviewed by managers. • (PR-02) Applications and operating systems must be patched routinely. (Bi-Monthly) • (PR-12) Only sys admins are permitted to install any software and/or hardware. • (PR-14) Perform information integrity checks on a routine basis, reviewed by DBA and/or managers. (Bi-Weekly) • (PR-20) Perform routine system and data backup. (Weekly) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly)

Number:	MC-16	
Name:	Input Validation Attack	
Scope:	System integrity concerns	
Priority:	___ Low ___ Medium _x_ High	
Deployment Environment:	_x_ Intranet ___ Extranet/Internet	
Mis-actors:	Users, sys admins	
Access Right Levels:	_x_ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User _x_ Sys Admin _x_ Other Network User	
Point of Entry:	___ Network ___ Host _x_ Application	
Security Attributes affected:	_x_ Confidentiality _x_ Integrity ____ Availability	
Description:	A user uses buffer overflow attacks or SQL injection attacks to gain unauthorized access to the system.	
Sophistication:	___ Low ___ Medium _x_ High	
Pre-conditions:	• The mis-actor has network access to the Asset Management System.	
Assumptions:		
Post-conditions:	Worst Case Threat:	• User gains unauthorized access to sensitive system data. His/her actions are never caught.
	Wanted Prevention Guarantee:	• Perform thorough input validation. • Hide HTML source code.
	Wanted Detection Guarantee:	• Audit information must be reviewed routinely. (Monthly)
	Wanted Recovery Guarantee:	N/A
Potential Mis-actor profiles:	Highly skilled users or sys admins with high criminal intent	
Stakeholders and Threats:	• AMS Client Company: loss of data confidentiality • Acme Group: loss of reputation, loss of current and potential clients	
Related Use Cases:	UC-01, UC-02, UC-03, UC-04, UC-05, UC-06, UC-07, UC-08	
Related Threats:	Buffer Overflow, SQL Injections, Query String Manipulation, Form Field Manipulation, Cookie Manipulation, HTTP Header Manipulation	

Architectural Recommendation:	• (AR-05) Check for buffer length. • (AR-16) Hide HTML source code. • (AR-33) Use least privileged account to access the database. • (AR-34) Use parameterized stored procedure for database access. • (AR-35) Use regular expressions to perform thorough input validation.
Policy Recommendation:	• (PR-02) Applications and operating systems must be patched routinely. (Bi-Monthly) • (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-04) Configuration changes are stored and cross-reviewed. (Monthly) • (PR-15) Perform routine code review. (Monthly) • (PR-20) Perform routine system and data backup. (Weekly)

Number:	MC-17
Name:	Infect Windows 2003 server with a virus or worm
Scope:	System integrity concerns
Priority:	___ Low ___ Medium _x_ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Users, sys admins
Access Right Levels:	_x_ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User _x_ Sys Admin _x_ Other Network User
Point of Entry:	___ Network _x_ Host ___ Application
Security Attributes affected:	___ Confidentiality ___ Integrity _x_ Availability
Description:	A user or sys admin sends a virus or worm to the server.
Sophistication:	___ Low ___ Medium _x_ High
Pre-conditions:	• The user's infected host resides on the same network.
Assumptions:	• Antivirus software is not installed or is not patched routinely.

Post-conditions:	Worst Case Threat:	• The server is infected by the virus or worm and fails to operate properly.
	Wanted Prevention Guarantee:	• A firewall is set up between servers and workstations. Antivirus software is installed on the server. • Patches are updated routinely. • IDS is installed on the network.
	Wanted Detection Guarantee:	• Log system and network resource usage.
	Wanted Recovery Guarantee:	• The server can be backed up from a previous image/backup.

Potential Mis-actor profiles:	Users or sys admins who may or may not have criminal intent.
Stakeholders and Threats:	• AMS Client Company: loss of data confidentiality • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-01, UC-02, UC-03, UC-04, UC-05, UC-06, UC-07, UC-08
Related Threats:	Denial of Service, Distributed Denial of Service, Worms/Trojans, Viruses, Malwares, Buffer Overflow, Toolkits/Rootkits

Architectural Recommendation:	• (AR-02) Install antivirus software on the server. • (AR-04) Block all unnecessary ports at the firewall and host. • (AR-09) Disable non-critical services and protocols. • (AR-15) Harden weak default configuration settings. • (AR-26) Set up firewalls with filtering rules between servers and workstations.
Policy Recommendation:	• (PR-02) Applications and operating systems must be patched routinely. (Bi-Monthly) • (PR-04) Configuration changes are stored and cross-reviewed. (Monthly) • (PR-05) Develop a disaster recovery contingency plan. • (PR-08) Firewalls and IDS must be patched routinely. (Monthly) • (PR-09) Follow the principle of least privilege and use least privileged service accounts to run processes and access resources. • (PR-10) Log all incoming and outgoing traffic (IIS, database engine, MapGuide, firewall). • (PR-11) New systems on the network should be evaluated prior to deployment. • (PR-20) Perform routine system and data backup. (Weekly)

Number:	MC-18
Name:	User gains access to the system using spoofed identities
Scope:	Network threat
Priority:	___ Low _x_ Medium ___ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Users, Sys Admins
Access Right Levels:	_x_ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User _x_ Sys Admin _x_ Other Network User
Point of Entry:	_x_ Network ___ Host ___ Application
Security Attributes affected:	_x_ Confidentiality _x_ Integrity ___ Availability
Description:	A user uses a fake source address to hide the original source of an attack or to work around network ACL that are in place to limit host based on source address rules.
Sophistication:	___ Low ___ Medium _x_ High
Pre-conditions:	• The user is on the same network as the Asset Management System.
Assumptions:	• The user eavesdrops on the traffic and figures out the data pattern.

Post-conditions:	Worst Case Threat:	• User gains unauthorized access to sensitive system data. His/her actions are never caught.
	Wanted Prevention Guarantee:	• Firewall with filtering rules prevents spoofed packets from originating in network. • Filter incoming packets that appear to come from an internal IP address at perimeter.
	Wanted Detection Guarantee:	• Log network resources and traffic (inbound and outbound) for unusual packets
	Wanted Recovery Guarantee:	N/A

Potential Mis-actor profiles:	Users or sys admins with high criminal intent.
Stakeholders and Threats:	• AMS Client Company: loss of data confidentiality • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-01, UC-02, UC-03, UC-04, UC-05, UC-06, UC-07, UC-08

Related Threats:	Password Cracking, Brute Force Attacks, Credential Theft, Cookie Replay Attacks, Network Eavesdropping, Dictionary Attacks
Architectural Recommendation:	• (AR-26) Set up firewalls with filtering rules between servers and workstations. • (AR-27) Set up an intrusion detection system (IDS).
Policy Recommendation:	• (PR-02) Applications and operating systems must be patched routinely. (Bi-Monthly) • (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-08) Firewalls and IDS must be patched routinely. (Monthly) • (PR-10) Log all incoming and outgoing traffic (IIS, database engine, MapGuide, firewall). • (PR-20) Perform routine system and data backup. (Weekly) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly)

Number:	MC-19
Name:	Information gathering/network eavesdropping
Scope:	Network Threat
Priority:	___ Low _x_ Medium ___ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Users, Sys Admins
Access Right Levels:	_x_ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User _x_ Sys Admin _x_ Other Network User
Point of Entry:	_x_ Network ___ Host ___ Application
Security Attributes affected:	_x_ Confidentiality _x_ Integrity ___ Availability
Description:	A user uses network footprinting devices or programs to illegally gather information about the network. (Examples: port scanning, intercept packets.)
Sophistication:	___ Low _x_ Medium _x_ High
Pre-conditions:	• The mis-actor is on the same network.
Assumptions:	

Post-conditions:	Worst Case Threat:	• A user identifies open ports, operating systems, applications, etc. that reside in the system. • A user steals an unencrypted data transfer. His/her actions are never caught.
	Wanted Prevention Guarantee:	• Configure routers to restrict system/workstations responses to footprinting requests. • Install software-based firewalls to prevent footprinting requests. • Disable any unused/vulnerable ports.
	Wanted Detection Guarantee:	• Log network resources and traffic (inbound and outbound) for footprinting requests.
	Wanted Recovery Guarantee:	N/A

Potential Mis-actor profiles:	Users or sys admins with high criminal intent
Stakeholders and Threats:	• AMS Client Company: loss of data confidentiality • Acme Group: loss of reputation, loss of current and potential clients

Related Use Cases:	UC-01, UC-02, UC-03, UC-04, UC-05, UC-06, UC-07, UC-08
Related Threats:	Sniffing, Footprinting, Port Scanning, Session Highjacking
Architectural Recommendation:	• (AR-04) Block all unnecessary ports at the firewall and host. • (AR-06) Configure routers to restrict footprinting requests. • (AR-09) Disable non-critical services and protocols. • (AR-20) Install software-based firewalls on all systems in the network. • (AR-25) Secure communication channels between servers and servers. • (AR-32) Use HTTPS for server-to-client Web data transfer encryption.
Policy Recommendation:	• (PR-02) Applications and operating systems must be patched routinely. (Bi-Monthly) • (PR-08) Firewalls and IDS must be patched routinely. (Monthly) • (PR-10) Log all incoming and outgoing traffic (IIS, database engine, MapGuide, firewall). • (PR-17) Routers must be patched routinely. (Monthly)

Number:	MC-20
Name:	Brute Force Attacks: Password Cracking/Credential Theft
Scope:	User authentication concerns
Priority	___ Low ___ Medium _x_ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Users, Sys Admins
Access Right Levels:	_x_ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User _x_ Sys Admin _x_ Other Network User
Point of Entry:	___ Network ___ Host _x_ Application
Security Attributes affected:	_x_ Confidentiality _x_ Integrity _____ Availability
Description:	A user tries to gain access to the system by cracking a user's account.
Sophistication:	___ Low ___ Medium _x_ High
Pre-conditions:	• The user is on the same network and can access the Asset Management System Web interface.
Assumptions:	• The Asset Management System allows unlimited user authentication retry.

Post-conditions:	Worst Case Threat:	• User credentials are stolen. • An attacker gains unauthorized access to sensitive system data using an assumed identity. His/her actions are never caught.
	Wanted Prevention Guarantee:	• Use strong passwords for all accounts. • Apply account lock-out policies. Limit the number of retry attempts. • Enforce users not to use operating system's default account names.
	Wanted Detection Guarantee:	• Perform failed logins for patterns of password hacking attempts.
	Wanted Recovery Guarantee:	N/A

Potential Mis-actor profiles:	Users or sys admins with high criminal intent
Stakeholders and Threats:	• AMS Client Company: loss of data confidentiality • Acme Group: loss of reputation, loss of current and potential clients

Related Use Cases:	UC-01, UC-02, UC-03, UC-04, UC-05, UC-06, UC-07, UC-08
Related Threats:	Spoofing, Session Hijacking, Password Cracking, Dictionary Attacks, Cookie Replay Attacks, Credential Theft
Architectural Recommendation:	• (AR-10) Display generic information on login screen (e.g., not "loosed-lipped"). • (AR-17) Implement account lock-out policies.
Policy Recommendation:	• (PR-02) Applications and operating systems must be patched routinely. (Bi-Monthly) • (PR-07) Enforce strong password policies. • (PR-09) Follow the principle of least privilege and use least privileged service accounts to run processes and access resources. • (PR-16) Require users to change their passwords periodically. (Monthly) • (PR-19) Set clear and defined user access controls for all users. (Low, Medium, High, System Admins) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly) • (PR-24) Users should not reveal their account names and passwords in any situation.

Number:	MC-21
Name:	Denial of Service
Scope:	Network Threat/System Unavailability
Priority:	___ Low ___ Medium _x_ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Users, Sys Admins
Access Right Levels:	_x_ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User _x_ Sys Admin _x_ Other Network User
Point of Entry:	_x_ Network ___ Host ___ Application
Security Attributes affected:	___ Confidentiality ___ Integrity _x_ Availability
Description:	A user disrupts services (application, software, hardware, and network) in the network, which causes system unavailability/downtime.
Sophistication:	___ Low ___ Medium _x_ High
Pre-conditions:	• The mis-actor is on the same network or has access to Asset Management System machines.
Assumptions:	• The mis-actor is capable of creating a large-scale DoS attack.

Post-conditions:	Worst Case Threat:	• Causes system/application/hardware downtime.
	Wanted Prevention Guarantee:	• Configure applications, services, and hardware with DoS in mind. • Apply patches and security updates regularly. • Harden TCP/IP stack against DoS/DDoS. • Invest in an intrusion detection system to detect intrusions and DoS/DDoS attacks. • Have redundant network and IT systems/hardware. • Back up systems regularly.
	Wanted Detection Guarantee:	• Review network traffic logs (firewalls, IDS, etc.) and system logs to detect malicious activity.
	Wanted Recovery Guarantee:	• Utilize system backups and redundant network systems/hardware in case of a DoS/DDoS attack.

Potential Mis-actor profiles:	Users or sys admins with high criminal intent.

Stakeholders and Threats:	• AMS Client Company: loss of data confidentiality • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-01, UC-02, UC-03, UC-04, UC-05, UC-06, UC-07, UC-08
Related Threats:	Distributed Denial of Service
Architectural Recommendation:	• (AR-21) Invest in backup IT hardware to ensure business continuity. • (AR-22) Invest in backup network capacity to avoid network downtime and system availability. • (AR-27) Set up an intrusion detection system.
Policy Recommendation:	• (PR-02) Applications and operating systems must be patched routinely. (Bi-Monthly) • (PR-05) Develop a disaster recovery contingency plan. • (PR-08) Firewalls and IDS must be patched routinely. (Monthly) • (PR-20) Perform routine system and data backup. (Weekly)

Number:	MC-22
Name:	Execute Malicious Code
Scope:	System integrity
Priority:	___ Low ___ Medium _x_ High
Deployment Environment:	_x_ Intranet ___ Extranet/Internet
Mis-actors:	Users, Sys Admins
Access Right Levels:	_x_ Low-Level System User _x_ Medium-Level System User _x_ High-Level System User _x_ Sys Admin _x_ Other Network User
Point of Entry:	___ Network ___ Host _x_ Application
Security Attributes affected:	_x_ Confidentiality _x_ Integrity _x_ Availability
Description:	A user gains unauthorized access to the system by executing malicious scripts on the server or on the client's workstation. Example: Improperly configured IIS servers may be exploited via canonicalization directory traversing or client may be exposed to cross-site scripting attacks.
Sophistication:	___ Low ___ Medium _x_ High
Pre-conditions:	• The mis-actor executes malicious code on the targeted/victim's computer. • The mis-actor plants the malicious code/script in the targeted computer.
Assumptions:	• The victim's machine is vulnerable to the malicious code (i.e., not patched). • The victim executes the malicious code/script (e.g., clicks on a Web link).

Post-conditions:		
	Worst Case Threat:	• Causes system downtime, data disclosure, unauthorized modification of data, etc.
	Wanted Prevention Guarantee:	• Perform thorough input validation.
	Wanted Detection Guarantee:	• Review system and application usage logs.
	Wanted Recovery Guarantee:	• System can be restored from a previous image/backup.

Potential Mis-actor profiles:	Users or sys admins with high criminal intent
Stakeholders and Threats:	• AMS Client Company: loss of data confidentiality • Acme Group: loss of reputation, loss of current and potential clients
Related Use Cases:	UC-01, UC-02, UC-03, UC-04, UC-05, UC-06, UC-07, UC-08

Related Threats:	Buffer Overflows, Cross Site Scripting, Canonicalization, Exceptions/Errors Reveal Implementation Details
Architectural Recommendation:	• (AR-13) Ensure that character encoding is set correctly to limit how input can be represented. • (AR-14) Handle and log exceptions that are allowed to propagate to the application boundary. • (AR-23) Keep custom configuration stores outside of the Web space. • (AR-24) Return generic, harmless error messages to the client. • (AR-30) Use exception handling through your application's code base. • (AR-31) Use HTMLEncode and URLEncode functions to encode any HTML output that includes user input. • (AR-35) Use regular expressions to make sure file names are well formed. • (AR-36) Use regular expressions to perform thorough input validation.
Policy Recommendation:	• (PR-02) Applications and operating systems must be patched routinely. (Bi-Monthly) • (PR-03) Audit information must be reviewed routinely. (Monthly) • (PR-04) Configuration changes are stored and cross-reviewed. (Monthly) • (PR-09) Follow the principle of least privilege and use least privileged service accounts to run processes and access resources. • (PR-20) Perform routine system and data backup. (Weekly) • (PR-21) User activities must be periodically reviewed. (Bi-Monthly)

Appendix F   Misuse Case Diagrams

Misuse Case Diagrams provide a graphical representation of the misuse cases. These diagrams identify which Asset Management System components were compromised and possibly affected by cyber-attacks.

Table of Contents

LEGEND

Application

Application Plug-in/ Component

Server

Database Management System

File(s)

Mis-actor/ Malicious user

Possibly Affected File(s)

Possibly Affected Server

Possibly Affected DMBS

Possibly Affected Application

Affected File(s)

Affected Server

Affected DBMS

Affected Application

Machine Border

Compromised Machine

Local Area Network Connection

Data Access Connection

File Access Connection

Control Transfer Connection

Callee Port

Caller Port

Read Access

Write Access

Read/ Write Access

MC-01: Unauthorized Logon on the Windows Server (2003 SE)
MC-18: User Gains Access to the System using Spoofed Identities

MC-02: System Administrator Gains Access to System Data

MC-03: Elevation of Privilege: Users Gain Sys Admin Rights

MC-04: Sys Admin Deletes Critical System Configurations

MC-05: Sys Admin Creates Holes in the System.

AMS User W2K Workstation
- Microsoft Internet Explorer (Ver 6)
- Microsoft Office 2000

Windows Server (2003 SE)

*.sdf files for MapGuide

MapGuide Author

MapGuide Ver (6.3)

*.dwg files for AutoCAD

Trojans/ Backdoors

Microsoft IIS (Ver 6.0)

EPP Documents *.asp, *.css, *.js, *.gif, *.jpg files for website

Sybase Central (Ver 4.1)

Creates holes in sys configuration

Sys Admin/Mis-Actor

AMS Developmental W2K Workstation

SDF Loader Program

Development Tools
- MS MapPoint
- MS Word 2000
- Edit Plus
- MenuMaker

AutoCad 2004 w/ Archibus Overlay

Archibus-FM (Ver 14.1)

MC-06: User Deletes Critical Data

AMS User W2K Workstation

Microsoft Internet Explorer (Ver 6)

Microsoft Office 2000

Step 1: Gain Access to workstation

Mis-Actor

Windows Server (2003 SE)

*.sdf files for MapGuide

Step 2: Using IE 6 to gain access to IIS/W2K3 Server

MapGuide Author

MapGuide Ver (6.3)

*.dwg files for AutoCAD

Microsoft IIS (Ver 6.0)

EPP Documents *.asp, *.css, *.js, *.gif, *.jpg files for website

Step 3: Deletes Critical Data

Sybase Central (Ver 4.1)

AMS Developmental W2K Workstation

SDF Loader Program

Development Tools

Edit Plus

MenuMaker

MS MapPoint   MS Word 2000

AutoCad 2004 w/ Archibus Overlay

Archibus-FM (Ver 14.1)

MC-07: User Falsifies Critical Data

AMS User W2K Workstation

Microsoft Internet Explorer (Ver 6)

Microsoft Office 2000

Step 1: Mis-Actor Gains Access to W2K Workstation

Mis-Actor

Windows Server (2003 SE)

*.sdf files for MapGuide

Step 2: Using IE 6 to gain access to IIS/W2K3 Server

MapGuide Author

MapGuide Ver (6.3)

*.dwg files for AutoCAD

Microsoft IIS (Ver 6.0)

EPP Documents *.asp, *.css, *.js, *.gif, *.jpg files for website

Step 3: Falsifies Critical Data

Sybase Central (Ver 4.1)

AMS Developmental W2K Workstation

SDF Loader Program

Development Tools

MenuMaker

Edit Plus

MS Word 2000

MS MapPoint

AutoCad 2004 w/ Archibus Overlay

Archibus-FM (Ver 14.1)

MC-08: Access System Data through Developmental Machines

AMS User W2K Workstation
- Microsoft Internet Explorer (Ver 6)
- Microsoft Office 2000

Windows Server (2003 SE)

EPP Documents *.asp, *.css, *.js, *.gif, *.jpg files for website

*.dwg files for AutoCAD

*.sdf files for MapGuide

MapGuide Author

MapGuide Ver (6.3)

Microsoft IIS (Ver 6.0)

Sybase Central (Ver 4.1)

Possible Attack

AMS Developmental W2K Workstation

SDF Loader Program

Development Tools

MS MapPoint   MS Word 2000

Edit Plus

MenuMaker

AutoCad 2004 w/ Archibus Overlay

Archibus-FM (Ver 14.1)

Possible Attack

Gains Access To Developmental Workstation

Sys Admin/Mis-Actor

8

MC-09: Access System Data Directly to/from Database

MC-10: Steal User Credentials Through Developmental Machines

MC-11: Unauthorized Viewing of System Data From Workstation

MC-12: Credential Theft Through Replay Attack
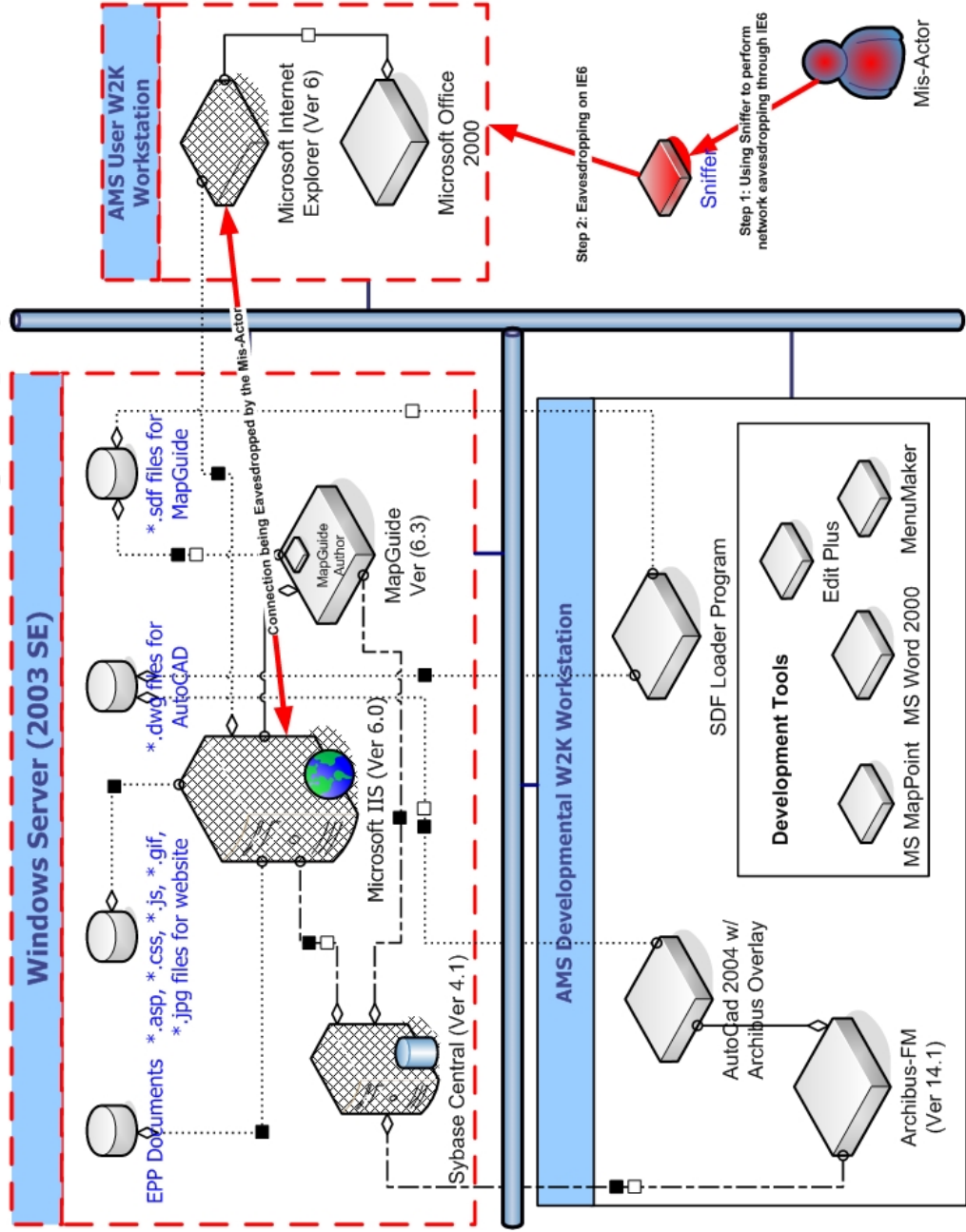
MC-13: Communications Tapped between Workstations and Server
MC-19: Information Gathering/Network Eavesdropping

MC-14: Unauthorized Access to Sensitive Data via Saved Excel Export Files on Victim's Machine.

MC-15: Malicious User Installs Malicious Programs to Tap into Excel's Memory to Steal Exported Data.

MC-16: Input Validation Attack

Windows Server (2003 SE)

AMS User W2K Workstation

Microsoft Internet Explorer (Ver 6)

Microsoft Office 2000

Step 1: Gain Access to system and Execute IE6

Mis-Actor

Step 1: Gain access to the system

EPP Documents *.asp, *.css, *.js, *.gif, *.jpg files for website

*.dwg files for AutoCAD

*.sdf files for MapGuide

Step 2: Performs IIS traversal attack or buffer overflow attack

MapGuide Author

MapGuide Ver (6.3)

Microsoft IIS (Ver 6.0)

Sybase Central (Ver 4.1)

Step 3: Performs SQL Injection Attack

AMS Developmental W2K Workstation

SDF Loader Program

Development Tools

Step 2: Gain Access to Archibus System

Edit Plus

MS MapPoint   MS Word 2000   MenuMaker

AutoCad 2004 w/ Archibus Overlay

Archibus-FM (Ver 14.1)

16

MC-17: Infects Server with Virus/Worms
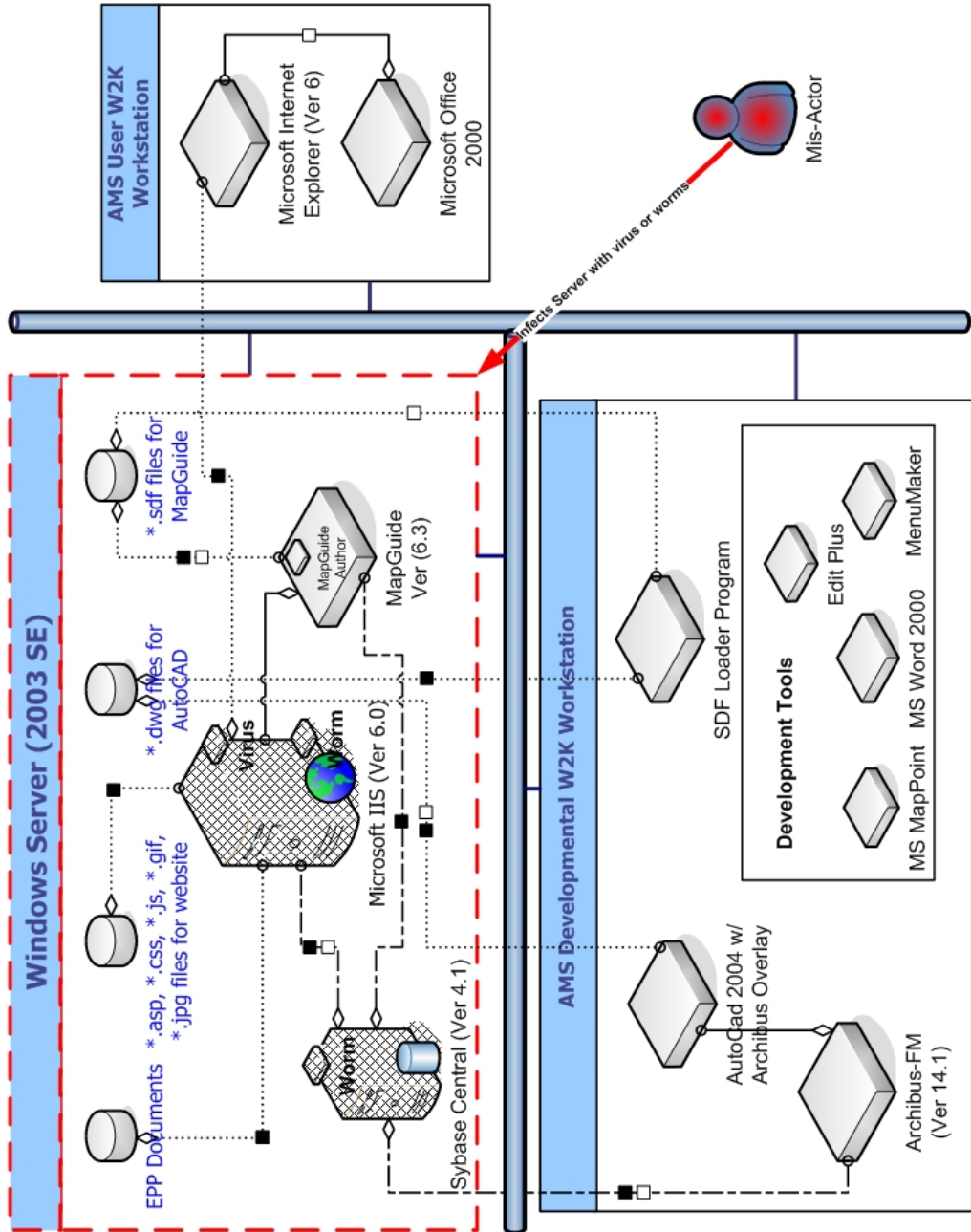
MC-20: Brute Force Attack: Password Cracking/Credential Theft

MC-21: Distributed/Denial of Service.

Windows Server (2003 SE)

EPP Documents *.asp, *.css, *.js, *.gif, *.jpg files for website

*.dwg files for AutoCAD

*.sdf files for MapGuide

MapGuide Author

MapGuide Ver (6.3)

Microsoft IIS (Ver 6.0)

Sybase Central (Ver 4.1)

AMS Developmental W2K Workstation

SDF Loader Program

Development Tools

MS MapPoint   MS Word 2000   MenuMaker

Edit Plus

AutoCad 2004 w/ Archibus Overlay

Archibus-FM (Ver 14.1)

AMS User W2K Workstation

Microsoft Internet Explorer (Ver 6)

Microsoft Office 2000

Step 2: Launches host specific DOS/DDOS

Step 2: Launches host specific DOS/DDOS

Step 2: Launches Network DOS/DDOS

Step 2: Launches host specific DOS/DDOS

DOS/DDOS Rootkit

Step 1: Mis-Actor utilizes rootkits to perform DOS/DDOS attacks

Mis-Actor

MC-22: Execute Malicious Code

Windows Server (2003 SE)

EPP Documents *.asp, *.css, *.js, *.gif,
*.jpg files for website

*.dwg files for
AutoCAD

*.sdf files for
MapGuide

Step 2: Launch IIS Traversal Attack

Microsoft IIS (Ver 6.0)

MapGuide
Author

MapGuide
Ver (6.3)

Sybase Central (Ver 4.1)

AMS User W2K Workstation

Microsoft Internet
Explorer (Ver 6)

Microsoft Office
2000

Step 1: Executes IE6

Mis-Actor

Executes Malicious Scripts

Executes Malicious Scripts

AMS Developmental W2K Workstation

SDF Loader Program

Development Tools

MS MapPoint   MS Word 2000

Edit Plus

MenuMaker

AutoCad 2004 w/
Archibus Overlay

Archibus-FM
(Ver 14.1)

20

Appendix G   Attack Tree Diagrams

Attack trees provide a formal, hierarchical way of describing the security of the system based on the types of attacks that could happen. These diagrams represent systems in a tree structure with the goal as the root node and tree leafs represent different ways to achieve that goal.

Table of Contents

Diagram Legend

LEGEND

And   Or   Scenario   Connector

MC-02: System Administrator Gains Access to System Data

- Exploit Application vulnerability (OR)
  - Buffer overflow
  - Virus
  - Worm
- Assume identity of an authorized user (OR)
  - Gain access to machine of authorized user (OR)
    - Replay in unclosed browser
    - Exploit poor account management (see MC-01)
    - Exploit poor password management (see MC-01)
  - Sys Admin create a user account
  - Sys Admin use other user account (OR)
    - Change password
    - Read password from database
  - (AND)
    - Hijack session
    - wiretap
- Directly Access database (OR)
  - Read unencrypted data
  - Elevate own rights
- steal data files

MC-13: Communications Tapped between Workstations and Server

- Exploit Application/OS/Network vulnerability
  - Replay command
  - Buffer overflow
  - Virus
  - Worm
- Install software sniffer
- Install hardware sniffer
  - Install device between input device and machine
  - Install hardware on the network
  - Steal network log
  - Obtain access to network log

MC-14: Unauthorized Access to Sensitive Data via Saved Excel Files on the Victim's Machine

- Access shared folders
  - See unprotected folders
  - Exploit poor password management
    - Use others' password
      - Steal password
        - Find written password
        - Install wiretap devices
      - Social engineering
    - Try blank password
    - Exploit easy-to-guess password
      - guess
      - Brute force
        - Toolkit
        - Dictionary attack
- Social engineering
- Assume identity of an authorized user
  - Gain access to machine of authorized user
    - Unclosed browser
    - Poor account management
    - Poor password management
- Exploit Application/OS vulnerability
  - Buffer overflow
  - Virus
  - Worm

Appendix H   Security Requirements

High-Priority-Level Misuse Cases with Corresponding ARs and PRs

Misuse Case 01	AR-01, AR-03, AR-19, PR-02, PR-03, PR-07, PR-13, PR-16, PR-19, PR-20, PR-21, PR-23, PR-24
Misuse Case 03	AR-03, AR-19, PR-02, PR-03, PR-04, PR-07, PR-13, PR-16, PR-20, PR-21, PR-22, PR-23, PR-24
Misuse Case 04	AR-03, PR-03, PR-04, PR-18, PR-20, PR-21
Misuse Case 06	AR-03, AR-19, PR-03, PR-04, PR-13, PR-14, PR-20, PR-21, PR-22, PR-23
Misuse Case 08	AR-08, AR-19, PR-03, PR-06, PR-07, PR-09, PR-10, PR-13, PR-14, PR-19, PR-20, PR-21, PR-23
Misuse Case 10	AR-12, AR-19, PR-03, PR-06, PR-07, PR-13, PR-14, PR-16, PR-19, PR-20, PR-21, PR-23, PR-24
Misuse Case 13	AR-25, AR-32, PR-01, PR-02, PR-08, PR-10, PR-12, PR-13, PR-21
Misuse Case 16	AR-05, AR-16, AR-33, AR-34, AR-36, PR-02, PR-03, PR-04, PR-15, PR-20
Misuse Case 17	AR-02, AR-04, AR-09, AR-15, AR-26, PR-02, PR-04, PR-05, PR-08, PR-09, PR-10, PR-11, PR-20
Misuse Case 20	AR-10, AR-17, PR-02, PR-07, PR-09, PR-16, PR-19, PR-21, PR-24
Misuse Case 21	AR-21, AR-22, AR-27, PR-02, PR-05, PR-08, PR-20
Misuse Case 22	AR-13, AR-14, AR-23, AR-24, AR30,

	AR-31, AR-35, AR-36, PR-02, PR-03, PR-04, PR-09, PR-20, PR-21

Architectural and Policy Recommendations
Table of Contents

Summary of Document

This document describes detailed information on the means of implementing a variety of architectural and policy recommendations using the current Asset Management System. These are implementations choices for the client to consider when solving any of these misuse cases, which were considered high priority by the client: MC-01, MC-03, MC-04, MC-06, MC-08, MC-10, MC-13, MC-16, MC-17, MC-20, MC-21, and MC-22. The goals refer to higher level objectives that the client desired to achieve and/or wished to implement. The security requirements attempt to narrow down the goals into rules or regulations containing security issues that may affect the system in which the client shall implement. Requirements are specific to the security, protection of data, and the system overall. The implementation choices identify methods in which the client can achieve these goals through either their existing system technologies or other technologies in the market.

The requirements were grouped into eight categories: Access Control, Encryption, Auditing, Privacy, Authentication, Survivability, Disaster Control, and Unauthorized Attack. The main purpose of the table is so that the client has a guide to find architectural and policy recommendation with ease regarding a specific topic. The purpose of the flow diagram is so when the client goes through the development process, they can trace from requirements down to implementation and back again, so that the client knows that every requirement has been implemented and that no extraneous functionality has been added. Most of the implementation choices were taken from Internet resources, such as Microsoft, Sybase, ARCHIBUS, SANS, CERT, SEI, and other security IT Web sites. For the goals and requirements, the Internet was also used as a resource, although most of the information was found through http://www.donald-firesmith.com. More detail regarding the specific resources can be found at the end of this document.

Architectural Security Requirements

AR-01

Goal(s):	The claimed identities of all users and client applications will be authenticated before they are allowed access.
	Protect from unauthorized attacks involving addition, modification, deletion, or replay of data in network
Category:	Authentication
Requirement(s):	AN-1) Authentication control mechanism shall be enforced in production environment. Authentication control will be done on user name and password or other user credentials.
No.	AR-01
Misuse case	MC-01
Architectural Recommendation	All shared drives on the network should enforce authentication policies.
Implementation Choices	In IIS 6.0, the IIS Manager contains a check box that permits the Administrator to omit the user name and password. If no user name and password are specified, IIS uses the requesting user credentials when the Administrator is using an authentication method that can perform delegation to authenticate to the remote share.

AR-02

Goal(s)	Protect network from unauthorized attacks.
Category:	Unauthorized Attacks
Requirement(s)	UA- 1) The system shall protect itself from viruses by using virus detection software with updated signatures. The virus detection software should also be run weekly.
No.	AR-02
Misuse case	MC-17
Architectural Recommendation	Antivirus software is installed on the server.
Implementation Choices	**Antivirus software mechanism using MS Server 2003** [Zamir 04] Antivirus software should have updated signatures files in order to function to its maximum potential. The Live Update feature requires both HTTP and FTP access to Symantec's Web site. In order to configure ISA to allow the main server to download definition updates, Administrators should use the following instructions: 1. Open the ISA management console. 2. Expand the *Server* -> *Policy Elements* -> *Client Address sets* in the ISA tree. 3. Create a Client address set named "NAV Server". Enter the IP address of the server on which the Norton AntiVirus Server is installed. If the NAV server is installed on the ISA Server itself (such as in the case of SBS 2000), make sure that the IP address specified is the internal IP address of the server (the private ISA server IP address). 4. Expand the *Access Policy* object, and create a new rule in *Protocol Rules*. This rule should allow the specified client address set that was created in step 3 to access FTP and HTTP sites. 5. Install Symantec Norton AntiVirus for Servers; include all the components recommended by Symantec. Open the "Norton AntiVirus Corporate Edition" program. From the File menu, select **Live Update**. Click on the **Configure** button, and move to the Proxy tab. Select "I want to customize my proxy settings for Live Update." Select both the **HTTP Proxy** and **FTP Proxy**. Enter the ISA Server's private network IP address (internal) for both Proxy settings. Use port 8080 for the HTTP proxy, and port 21 for the FTP proxy. Apply the changes and click **OK**. Click **Next**, and then test the ability to download the required updates. 6. To schedule a daily download of new virus definitions, open the "Symantec System Center" console, expand the "System Hierarchy," and right-click the Norton Server group (by default named "Norton AntiVirus 1"). Select *All Tasks* -> *Norton AntiVirus* -> *Virus Definition*

	Manager….
	In the *Virus Definition Manager* screen, select *Update the primary Server Group only*.
	In *How servers retrieve virus definitions updates*, click **Configure**.
	Select the required schedule for the definition download.
	In order to check whether the settings work, click **Update Now**.
	If the update was successful, an event ID 16 should be logged in the server's application log, which will indicate if the definitions are current (was able to connect, but no download was required), or if the download of the virus definition file was successful.
	Note! There is no need to configure the Live Update proxy use from the Symantec System Center.
	7. Configure the client computer to retrieve the definition update. Update virus definitions from parent server. The new virus definitions should be downloaded by the NAV server.

AR-03

Goal(s):	Enforce audit mechanisms to detect unauthorized users and to support future incident investigations. Ensure that the application or component collects, analyzes, and reports information about • all security-related events • the status (e.g., enabled vs. disabled, updated versions) of its security mechanisms • the use of its security mechanisms (e.g., access and modification by security personnel) Ensure that the application or component collects sufficient information regarding potential breaches of security to establish what events occurred, when they occurred, and who (or what) caused them. Enable security personnel to audit the status and usage of the security mechanisms.
Category:	Auditing
Requirement(s):	AU-1) The system shall audit systems on network and user logging information. This shall be put into practice monthly.
No.	AR-03
Misuse case	MC-01, MC-02, MC-03, MC-04, MC-05, MC-06, MC-07
Architectural Recommendation	Audit information is stored in a separate location from the servers and the workstation.
Implementation Choices	**Microsoft Server 2003** [Microsoft 03a] According to the Microsoft Web site, information can be audited by using Microsoft Windows Server 2003. If an intrusion occurs, Microsoft recommends isolating and preserving the security log entries. These entries can be valuable during an investigation of the intrusion. An audit trail can contain information about changes that are made to the computer or to other computers on the network. Microsoft Operations Manager is an example of a tool that regularly collects and saves security log entries across the organization. Even if intruders or administrators clear the local security log, the Administrator is more likely to be able to trace their actions. Microsoft also recommends deciding what type of information the Administrator may want to gain by collecting audit events. If, for example, the Administrator is interested in tracking the attempts of users to gain access to areas for which they are not authorized, the Administrator can collect failure audits Overall the organization should consider the resources that are available for collecting and reviewing audit log files. **Microsoft Internet Security and Acceleration (ISA) Server** [Shinder 04a] Logging should always be enabled for each service. This is the default setting. Never disable logging for any of the services because the logs are the primary information source in determining the origin of a major attack or troubleshooting

a problem with access and access controls.

New log files should be created each day, and the log files should be copied from the ISA Server to a safe location each day so that they are available if a hacker or hardware failure makes it impossible to retrieve them. Save the logs in **ISA Server file format.**

The more fields that are logged, the more system resources will be required for logging. Review the fields included in the log files.

ISA Server is preconfigured to save the last 30 log files. The daily creation of new log files will allow the storage of one month's worth of log files. Acme may want to increase the number of log files to a larger number, possibly a year's worth. Enable the compression option for the log files so that they do not consume excessive disk space. The SQUARE team highly recommends that Acme get at least a 100+ GB drive dedicated to the storage of Acme's log files. With compression, it is possible to store approximately 200 GB of log files.

The ISA Server reports are constructed using Log Summaries. Select the Enable Reports option to create the reports. Select the **Enable daily and monthly summaries** options.

Main Points

Store Logs and Summaries on a dedicated, extendable disk.

Increase the number of saved log files.

Copy the log files each day to a safe location.

Increase the number of saved summaries.

AR-04, AR-09

Goal(s)	The claimed identities of all users and client applications will be authenticated before they are allowed access. Protect from unauthorized attacks involving addition, modification, deletion, or replay of data in network.
Category:	Authentication
Requirement(s)	AN-2) Authentication control mechanism shall be enforced in production environment. Authentication control will be done on the network level.
No.	AR-04, AR-09
Misuse case	MC-17, MC-19,
Architectural Recommendation	Disable non-critical services and protocols and block all unnecessary ports at the firewall and host.
Implementation Choices	Principle of least privilege. Only allow outbound access for protocols that are required. Also, only permit those users who require access to those protocols the use of said protocols. If there is no business need for a particular protocol, do not allow access to it. Survey the users for the applications they require to get their jobs done. Confer with the security team in Acme to determine which protocols are required by the organization. After Acme determines the required protocols and who needs to use them, create the appropriate protocol rules. Periodically review the firewall, packet filter, and Web proxy logs. Acme's review will be helpful in determining whether users are attempting to use unapproved applications. The firewall logs are especially helpful in this regard, as applications report to the firewall client and service their names [Shinder 04a].

AR-05

Goal(s):	Protect network from unauthorized attacks.
Category:	Unauthorized Attacks
Requirement(s):	UA-2) The system shall be able to determine when a buffer overflow attack has occurred. If attack takes place, system should shut down and system administrator should be notified within a reasonable time.
No.	AR-05
Misuse case	MC-16
Architectural Recommendation	Check for buffer length.
Implementation Choices	**Buffer and memory-overflow protection feature in IIS server** [Microsoft 03b]**:** IIS 6.0 assists in resisting a common method of attack on Web servers: buffer and memory overflow situations. An attacker can penetrate a server by taking advantage of the way a Web server processes data transmissions of unknown size. IIS 6.0 closes this vulnerability with memory-overflow protection, which helps ensure that once a buffer or memory overflow has been detected in a particular process, the process will be shut down so that it cannot affect other processes within the system. **Buffer feature through Sybase** [Sybase 97]**:** The ASE Plug-in for Sybase Central includes a variety of monitor abilities. Device I/O Monitor shows buffer (not page) I/O activity on the database devices defined for ASE. Network Activity Monitor displays packet volume and packet sizes used for communication between ASE and its clients. The monitor also shows values of some configuration parameters that affect network traffic. ASE release 11.5.1 (a maintenance release) will include an additional monitor, the Process Current SQL Statement Monitor, which displays the SQL statement and query plan currently executing in a selected process.

AR-08

Goal(s):	All users and client applications will be identified before they are allowed access.
	Protect from unauthorized attacks involving addition, modification, deletion, or replay of data in network.
Category:	Access Control
Requirement	AC-1) The system shall not grant any user access to one or more system services and data during a secure session without first identifying the user.
	AC-2) The system shall not grant any client application access to one or more system services and data during a secure session without first identifying the client application.
	AC-3) If no identification is provided, the system shall record the security event and notify the operator within a reasonable time.
No.	AR-08
Misuse case	MC08
Architectural Recommendation	Developmental machines should have a strong access control mechanism.
Implementation Choices	**Access control through IIS** [Microsoft 04a]
	Access control can be accomplished though the IIS application by using these countermeasures, which include secure Web permissions, NTFS permissions, and .NET Framework access control mechanisms including URL authorization.
	The following steps outline the access-control process:
	1. The client requests a resource on the server.
	2. The IP address of the client is checked against any IP address restrictions in IIS. If the IP address is denied access, the request fails, and a "403 Access Forbidden" message is returned to the user.
	3. The server, if configured to require it, requests authentication information from the client. The browser either prompts the user for a user name and password or offers this information automatically.
	4. IIS checks whether the user has a valid Windows user account. If the user does not, the request fails, and a "401 Access is denied" message is returned to the user.
	5. IIS checks whether the user has Web permissions for the requested resource. If the user does not, then the request fails, and a "403 Access Forbidden" message is returned to the user.
	6. Any security modules, such as ASP.NET impersonation, are added.
	7. IIS checks the NTFS permissions on static files, ASP, and Common CGI files for the resource. If the user does not have NTFS permissions for the resource, then the request fails, and a "401 Access is denied" message is returned to the user.
	8. If the user has NTFS permissions, the request is fulfilled.
	Access Control through Sybase [Sybase 03]
	Access control can be facilitated through the Sybase Application by using the ASE plug-

in. The ASE plug-in has security features that include Role-Based Access Control, proxy authorization, single sign-on, and a C2 certification. ASE includes a Policy-Based Access Control framework that provides a means of protecting data down to the row level. Administrators can define security policies that are based on the value of individual data elements. The server enforces these policies. Once a policy has been defined, it is automatically invoked whenever the affected data is queried, whether through an application, ad hoc query, stored procedure, or view. This is accomplished through the four combined capabilities of

- access rules
- the Application Context Facility
- login triggers
- domain integrity rules

Access control through current version ARCHIBUS 14.1 [ARCHIBUS 02]

Integrating ARCHIBUS/FM with AutoManager Meridian can help with access control and change management to CAD drawings.

Security using newer ARCHIBUS version 14.2 [ARCHIBUS 04]

An option would be to map hierarchical security groups to roles that reflect the existing data or organizational divisions.

- Control information access based on user type; grant security levels to multiple users with similar job responsibilities.
- Coordinate ARCHIBUS/FM restrictions with SQL-level database security.
- Assign a process or role, such as program access, to a staff member.
- Define and coordinate the roles of the entire staff, including contractors and/or vendors.
- Enforce data security by restricting access to sensitive information or restricting approvals based on user login.

AR-10, AR-17

Goal(s)	Data and communications shall not be corrupted.
	Ensure that persons understand and have reasonable control over their private information, thereby minimizing potential bad press and loss of user confidence.
Category:	Privacy
Requirement(s)	PV-1) The system shall not permit any user login data to be retrieved by an attacker.
No.	AR-10, AR-17
Misuse case	MC-20
Architectural Recommendation	Display generic information on login screen (e.g., not loosed-lipped)
	Implement account lock-out policies.
Implementation Choices	**Setting a title for the login message** Double-click **Interactive Logon: Message title for users attempting to log on**. Type in the desired massage to display in the title bar. **Disable showing the last user at the login screen** Double-click **Interactive Logon: Do not display last user name**. Set the radio button to **Enabled**. **Setting a message to show up in a dialog box after users press Ctrl-ALT-DEL at the login screen** Double-click **Interactive Logon: Message text for users attempting to log on**. Type in the desired message to be displayed and press **OK**. **Setting Account Lockout Durations** Click **Start**, and then **Run**. In the **Run** box, type "gpedit.msc." Under **Computer Configuration**, click the "+" next to **Windows Settings**, then **Security Settings**, then **Account Policy**, and then **Account lockout**. Double-click on **Account lockout threshold** and enter the desired **max log-in attempt**. Click **OK**. A dialog box will display a message indicating that Windows will enable two other items with recommended settings. Click **OK**. Double-click **Account lockout duration**. This will be the amount of time after unsuccessful logins that the account will be locked for. Put in the value Acme would like and press **OK**. Double-click **Reset account lockout counter after:**. This is the amount of time Acme wants Windows Server 2003 to remember invalid logins for lockout. Click the user's folder and then double-click the locked out user. The Administrator will see a checkbox checked by **Account is locked out**. Unchecking that will unlock the account.

AR-12

Goal(s):	Ensure the security and confidentiality of user records and information.
	Protect against any anticipated threats or hazards to the security or integrity of user credentials.
	Ensure that confidential communications and data are kept private.
Category:	Encryption
Requirement(s):	EN-1) The system's data and communication shall be encrypted.
No.	AR-12
Misuse case	MC10
Architectural Recommendation	Encrypt user credentials in configuration and databases.
Implementation Choices	**Encryption though IIS** [Microsoft 04b] The following demonstrates how to implement forms-based authentication by using a database to store the user's credentials. Notes • Acme can use the **FormsAuthentication** class utility function named **HashPasswordForStoringInConfigFile** to encrypt the passwords before they are stored in the database or configuration file. • Acme may want to store the SQL connection information in the configuration file (Web.config) so that it is modifiable if necessary. Another consideration would be to add code to prevent hackers from using various username/password combinations and logging on. For example, it is possible to include logic that accepts only two or three logon attempts. If the user cannot log on in a certain number of attempts, Acme may want to set a flag in the database to not allow that user to log on until that user re-enables his or her account by visiting a different page or by calling the support line. • Acme should add appropriate error handling wherever necessary. Because the user is identified based on the authentication cookie, Acme may want to use Secure Sockets Layer (SSL) on this application so that no one can deceive the authentication cookie and any other valuable information that is being transmitted. • Forms-based authentication requires that clients accept or enable cookies on their browser. • The **timeout** parameter of the **<authentication>** configuration section controls the interval at which the authentication cookie is regenerated. Select a value that provides enhanced security. • Certain intermediary proxies and caches on the Internet may cache Web server responses that contain Set-Cookie headers, which are then returned to a different user. Because forms-based authentication uses a cookie to authenticate users, this can cause users to accidentally (or intentionally) impersonate another user by receiving a cookie from an intermediary proxy or cache that was not originally intended for them.

	Encryption through Sybase
	Encrypting information can be accomplished though the Sybase Application by using the ASE plug-in that employs certified SSL encryption algorithms to protect data as it is transferred between the database server and its clients.

Goal(s)	Protect network from unauthorized attacks.
Category:	Unauthorized Attacks
Requirement(s)	UA-3) The system shall protect itself from malicious encoding mechanisms.
No.	AR-13, AR-14, AR-23, AR-24, AR-30, AR-31
Misuse case	MC22
Architectural Recommendation	Use HTML Encode and URL Encode functions to encode any HTML output that included user input.
	Ensure that character encoding is set correctly to limit how input can be represented.
	Keep custom configuration stores outside of the Web space.
	Use exception handling through the application code's base.
	Handle and log exceptions that are allowed to propagate to the application boundary.
	Return generic, harmless error message to the client.
Implementation Choices	**Encode/Decode** According to *Using Server.URLEncode*, a method to resolve the above architectural recommendation and the problem of having other non-alphanumeric characters in the URL/querystring is through encoding and decoding mechanism [Humpherys 04]. The author suggests using **Server.UrlEncode**, which will successfully map those "illegal" non-alphanumeric characters into the correct ASCII codes and also for decoding. Use **Request.Querystring**; the conversion from the ASCII equivalent of non-alphanumeric characters *back* to their original character form will occur automatically. **ASP Built-in Objects** [Microsoft 04c] This section describes the intrinsic COM objects (ASP built-in objects) that are available to ASP pages. Using ASP built-in objects, it is possible to access information regarding the Web server, the client who is accessing a Web page, the Web application that contains the Web page, and the fields in the HTTP request and response streams. The ASP built-in objects are organized by the type of information they contain. The information in ASP built-in objects can also be obtained in a COM component or an ISAPI application. The following table lists the technologies from which ASP built-in objects can be accessed and how to access them. {{TABLE}}

Technology	Method of accessing ASP built-in objects
ASP	Use the ASP built-in objects listed in this section.
ASP.NET	The Request, Response, Server, Application, and Session objects are part of ASP.NET and are used in much the same

	way as they are in ASP. However, in ASP.NET these objects are defined in new classes in the System.Web namespace. For more information, see ASP.NET Intrinsic Objects.
COM component	Use the C++ interfaces, the Java classes, or the COM+ **ObjectContext** object by calling **GetObjectContext** to gain access the ASP built-in objects.
ISAPI application	Use the C++ interfaces for the ASP built-in objects.

This section includes the following topics:

Topic	Description
Application Object	Describes the methods, properties, and collections of the object that stores information related to the entire Web application, including variables and objects that exist for the lifetime of the application.
ASPError Object	Describes the properties of the object that stores information about an error condition.
ObjectContext Object	Describes a wrapper for the COM+ **ObjectContext** object, which provides methods and events that are used only for transaction processing.
Request Object	Describes the methods, properties, and collections of the object that stores information related to the HTTP request. This includes forms, cookies, server variables, and certificate data.
Response Object	Describes the methods, properties, and collections of the object that stores information related to the server's response. This includes displaying content, manipulating headers, setting locales, and redirecting requests.
ScriptingContext Object	In a component, the **ScriptingContext** object returns references to the ASP built-in objects; however, this is an obsolete and unsupported method, removed in IIS 4.0. Use the COM+ **ObjectContext** object to return references to the ASP built-in objects.
Server Object	Describes the methods and properties of the object that provides methods for various server tasks. With these methods the Administrator can execute code, get error conditions, encode text strings, create objects for use by the Web page, and map physical paths.

		Session Object	Describes the methods, properties, and collections of the object that stores information related to the user's session, including variables and objects that exist for the lifetime of the session.

Goal(s)	Protect network from unauthorized attacks.
Category:	Unauthorized attacks
Requirement(s)	UA-4) The system shall protect itself from attacks due to weak server configuration.
No.	AR-15
Misuse case	MC-17
Architectural Recommendation	Harden weak default configuration setting.
Implementation Choices	**Securing the Web Server** According to an article titled "Securing Your Web Server," a secure Web server provides a solid foundation for the hosting environment, where its configuration plays a critical role in the overall security of Web applications [Microsoft 04d]. This module applies to the following products and technologies: • Microsoft® Windows® Server 2000 and Windows Server™ 2003 operating systems • Microsoft .NET Framework 1.1 and ASP.NET 1.1 • Microsoft Internet Information Services (IIS) 5.0 and 6.0 **Methodology for Securing Acme's Web Server** To secure the Web server, Acme must apply many configuration settings to reduce the server's vulnerability to attack. **Configuration Categories** The security methodology in this module has been organized into the categories shown in Figure 2.
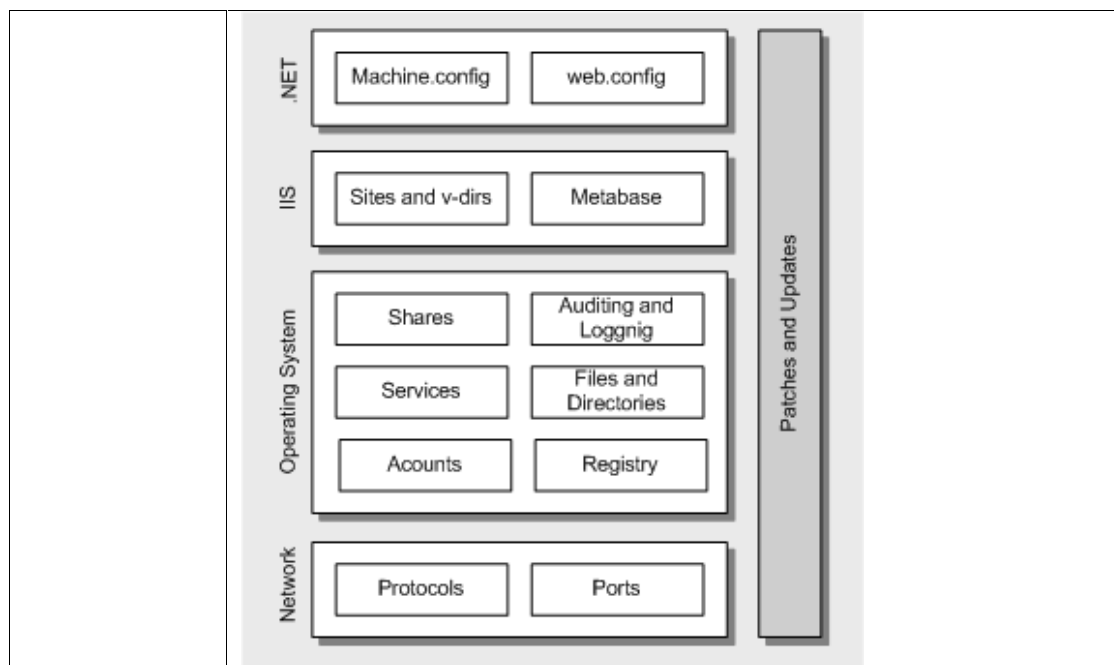
Figure 2. Web Server Configuration Categories

The rationale behind the categories is as follows:

Patches and Updates

Patching and updating the server software is a critical first step towards securing the Web server.

Services

Services are prime vulnerability points for attackers, who can exploit the privileges and capabilities of a service to access the local Web server or other downstream servers. If a service is not necessary for the Web server's operation, do not run it. If the service is necessary, secure it and maintain it. Consider monitoring any service to ensure availability. If the service software is not secure, but the service is still needed, try to find a secure alternative.

Protocols

Avoid using protocols that are inherently insecure. If the Administrator cannot avoid using these protocols, take the appropriate measures to provide secure authentication and communication (by using IPSec policies, for example). Examples of insecure, clear text protocols are Telnet, Post Office Protocol (POP3), Simple Mail Transfer Protocol (SMTP), and File Transfer Protocol (FTP).

Accounts

Accounts must be audited. Configure accounts with least privilege to help prevent elevation of privilege. Remove any accounts that are not needed. Slow down brute force and dictionary attacks with strong password policies, and then audit and alert for logon failures.

Files and Directories

Secure all files and directories with restricted NTFS permissions that only allow

access to necessary Windows services and user accounts. Use Windows auditing to detect suspicious or unauthorized activity.

Shares

Remove all unnecessary file shares, including the default administration shares if they are not required. Secure any remaining shares with restricted NTFS permissions.

Ports

Audit the ports on the server regularly to ensure that an insecure or unnecessary service is not active on the Web server. An active port that was not opened by an administrator is a sure sign of unauthorized access and a security compromise.

Registry

The Administrator must secure the registry. Apply restricted Windows ACLs and block remote registry administration.

Auditing and Logging

Use a combination of Windows and IIS auditing features to configure auditing on the server.

Sites and Virtual Directories

Relocate sites and virtual directories to non-system partitions and use IIS Web permissions to further restrict access.

Script Mappings

Remove all unnecessary IIS script mappings for optional file extensions to prevent an attacker from exploiting any bugs in the ISAPI extensions that handle these types of files. Unused extension mappings are often overlooked and represent a significant security vulnerability.

ISAPI Filters

Attackers have been successful in exploiting vulnerabilities in ISAPI filters. Remove unnecessary ISAPI filters from the Web server.

IIS Metabase

The IIS metabase maintains IIS configuration settings. Ensure that the security-related settings are appropriately configured and that access to the metabase file is restricted with hardened NTFS permissions.

Code Access Security

Restrict code access security policy settings to ensure that code downloaded from the Internet or intranet has no permissions and as a result will not be allowed to execute.

IIS Installation Considerations

What Does IIS Install?

IIS installs a number of services, accounts, folders, and Web sites. Some components that IIS installs may not be used by Acme's Web applications, and if present on the server could make the server vulnerable to attack.

Table H-1. IIS Installation Defaults

Item	Details	Default
Services	IIS Admin Service (for administration of Web and FTP services)	Installed
	World Wide Web Publishing Service	Installed
	FTP Publishing Service	Installed
	Simple Mail Transport Protocol (SMTP)	Installed
	Network News Transport Protocol (NNTP)	Installed
Accounts and Groups	IUSR_MACHINE (anonymous Internet users)	Added to Guest group
	IWAM_MACHINE (out-of-process ASP Web applications; not used for ASP.NET applications except those running on a domain controller; the Web server should not be a domain controller)	Added to Guest group
Folders	%windir%\system32\inetsrv (IIS program files) %windir%\system32\inetsrv\iisadmin (Files used for remote IIS admin) %windir%\help\iishelp (IIS help files) %systemdrive%\inetpub (Web, FTP, and SMTP root folders)	
Web Sites	Default Web Site–port 80: %SystemDrive%\inetpub\wwwroot Administration Web Site–port 3693: %SystemDrive%\System32\inetsrv\iisadmin	Anonymous access allowed Local machine and Administrators access only

Installation Recommendations

The recommendation is that Acme does not install IIS as part of the operating system installation but that they install it later, after they have updated and patched the base operating system. After the install of IIS, reapply IIS patches and harden the IIS configuration to ensure that it is fully secured. Only then is it safe to connect the server to the network.

Steps for Securing the Web Server

The next sections guide the Administrator through the process of securing the Web server.

Step 1. Patches and Updates

Update the server with the latest service packs and patches. Update and patch all of the Web server components, including Windows 2003 (and IIS) and Microsoft Data Access Components (MDAC).

During this step:

Detect and install the required patches and updates.

Step 2. IISLockdown

The IISLockdown tool helps to automate certain security steps. IISLockdown reduces the vulnerability of a Windows 2003 Web server. It allows for picking a specific type of server role and then using custom templates to improve security

for that particular server. The templates either disable or secure various features. In addition, IISLockdown installs the URLScan ISAPI filter. URLScan allows Web site administrators to restrict the kind of HTTP requests that the server can process, based on a set of rules that the administrator controls. By blocking specific HTTP requests, the URLScan filter prevents potentially harmful requests from reaching the server and causing damage.

During this step:

- Install and run IISLockdown.
- Install and configure URLScan.

Step 3. Services

Services that do not authenticate clients, services that use insecure protocols, and services that run with too much privilege are risks. If they are not needed, disable them. Disabling unnecessary services quickly and easily reduces the attack surface. Overhead, in terms of maintenance (patches, service accounts, and so on), is also reduced. Ensure that all services are secure and maintained. Run the service using a least privilege account, and keep the service current by applying patches.

During this step:

- Disable unnecessary services.
- Disable FTP, SMTP, and NNTP unless Acme requires them.
- Disable the ASP.NET State service unless Acme requires it.

Step 4. Protocols

By preventing the use of unnecessary protocols, the potential for attack is decreased. The .NET Framework provides granular control of protocols through settings in the Machine.config file. For example, an Administrator can control whether the Web Services can use HTTP GET, POST or SOAP.

During this step:

- Disable or secure WebDav.
- Harden the TCP/IP stack.
- Disable NetBIOS and SMB.

Step 5. Accounts

Remove accounts that are not used, because an attacker might discover and use them. Require strong passwords. Weak passwords increase the likelihood of a successful brute force or dictionary attack. Use least privilege. An attacker can use accounts with too much privilege to gain access to unauthorized resources.

During this step:

- Delete or disable unused accounts.
- Disable the Guest account.
- Rename the Administrator account.
- Disable the IUSR account.
- Create a custom anonymous Web account.
- Restrict remote logons.
- Disable null sessions (anonymous logons).

Delete or Disable Unused Accounts

Unused accounts and their privileges can be used by an attacker to gain access to a server. Audit local accounts on the server and disable those that are unused. If disabling the account does not cause any problems, delete the account. (Deleted accounts cannot be recovered.) Disable accounts on a test server before the Administrator disables them on a production server. Make sure that disabling an account does not adversely affect the application operation.

Note The Administrator account and the Guest account cannot be deleted.

Disable the Guest Account

The Guest account is used when an anonymous connection is made to the computer. To restrict anonymous connections to the computer, keep this account disabled. The guest account is disabled by default on Windows 2003. To check whether or not it is enabled, display the **Users** folder in the Computer Management tool. The Guest account should be displayed with a cross icon. If it is not disabled, display its **Properties** dialog box and select **Account is disabled**.

Rename the Administrator Account

The default local Administrator account is a target for malicious use because of its elevated privileges on the computer. To improve security, rename the default Administrator account and assign it a strong password.

If Acme intends to perform local administration, configure the account to deny network logon rights and require the administrator to log on interactively. By doing so, Acme prevents users (well intentioned or otherwise) from using the Administrator account to log on to the server from a remote location. If a policy of local administration is too inflexible, implement a secure remote administration solution.

Disable the IUSR Account

Disable the default anonymous Internet user account, IUSR_MACHINE. This is created during IIS installation. MACHINE is the NetBIOS name of the server at IIS installation time.

Create a Custom Anonymous Web Account

If Acme's applications support anonymous access (for example, because they use a custom authentication mechanism such as Forms authentication), create a custom least privileged anonymous account. If Acme runs IISLockdown, add the custom user to the Web Anonymous Users group that is created. IISLockdown denies access to system utilities and the ability to write to Web content directories for the Web Anonymous Users group.

If Acme's Web server hosts multiple Web applications, Acme may want to use multiple anonymous accounts, one per application, so that it can secure and audit the operations of each application independently.

Enforce Strong Password Policies

To counter password guessing and brute force dictionary attacks on the application, apply strong password policies. To enforce a strong password policy:

Set password length and complexity. Require strong passwords to reduce the threat of password guessing attacks or dictionary attacks. Strong passwords are

eight or more characters and must include both alphabetic and numeric characters.

Set password expiration. Passwords that expire regularly reduce the likelihood that an old password can be used for unauthorized access. Frequency of expiration is usually guided by a company's security policy.

Table H-2 shows the default and recommended password policy settings.

Table H-2. Password Policy Default and Recommended Settings

Password Policy	Default Setting	Recommended Minimum Setting
Enforce password history	1 password remembered	24 passwords remembered
Maximum password age	42 days	42 days
Minimum password age	0 days	2 days
Minimum password length	0 characters	8 characters
Passwords must meet complexity requirement.	Disabled	Enabled
Store password using reversible encryption for all users in the domain.	Disabled	Disabled

Restrict Remote Logons

Remove the Access this computer from the network privilege from the Everyone group to restrict who can log on to the server remotely.

Disable Null Sessions (Anonymous Logons)

To prevent anonymous access, disable null sessions. These are unauthenticated or anonymous sessions established between two computers. Unless null sessions are disabled, an attacker can connect to the server anonymously (without being authenticated).

Once an attacker establishes a null session, he or she can perform a variety of attacks, including enumeration techniques used to collect system-related information from the target computer—information that can greatly assist subsequent attacks. The type of information that can be returned over a null session includes domain and trust details, shares, user information (including groups and user rights), registry keys, and more.

Restrict Null sessions by setting RestrictAnonymous to 1 in the registry at the following subkey:

    HKLM\System\CurrentControlSet\Control\LSA\RestrictAnonymous=1

Additional Considerations

The following is a list of additional steps Acme can consider to further improve security on the Web server:

- Require approval for account delegation. Do not mark domain accounts in Active Directory as trusted for delegation unless the Administrator first obtains special approval to do so.

- Do not use shared accounts. Do not create shared accounts for use by multiple individuals. Authorized individuals must have their own accounts. The activities of individuals can be audited separately and group membership and privileges appropriately assigned.
- Restrict the local administrators group membership. Try to limit administration accounts to two. This helps provide accountability. Also, passwords must not be shared, again to provide accountability.
- Require the Administrator to log on interactively. If there is local administration only, require the Administrator account to log on interactively by removing the Access this computer from the network privilege.

Step 6. Files and Directories

Install Windows 2003 on partitions formatted with the NTFS file system so that there is a benefit from NTFS permissions to restrict access. Use strong access controls to protect sensitive files and directories. In most situations, an approach that allows access to specific accounts is more effective than one that denies access to specific accounts. Set access at the directory level whenever possible. As files are added to the folder they inherit permissions from the folder, so the Administrator needs to take no further action.

During this step:

- Restrict the Everyone group.
- Restrict the anonymous Web account(s).
- Secure or remove tools, utilities, and SDKs.
- Remove sample files.

Restrict the Everyone Group

The default NTFS permissions for Windows 2003 grant members of the Everyone group full control access to a number of key locations, including the root directory, \inetpub, and \inetpub\scripts.

First grant FULL CONTROL to the Administrator account to the root (\), then remove access rights for the Everyone group from the following directories:

- root (\)
- system directory (\WINNT\system32)
- Framework tools directory (\WINNT\Microsoft.NET\Framework\{version})
- Web site root directory and all content directories (the default is \inetpub*)

Restrict Access to the IIS Anonymous Account

Attackers target this well-known account to perform malicious actions. To secure the anonymous account:

- Deny write access to Web content directories. Make sure that it is not possible for this account to write to content directories (to deface Web sites, for example).
- Restrict access to System tools. Restrict access to command-line tools located in \WINNT\System32.
- Assign permissions to groups instead of individual accounts. Assigning users to groups and applying permissions to groups instead of individual

accounts is good practice. For the anonymous account, create a group and add the anonymous account to it and then explicitly deny access to the group for key directories and files. Assigning permissions to a group allows the Administrator to more easily change the anonymous account or create additional anonymous accounts because they do not need to recreate the permissions.

Note IISLockdown denies write access to content directories for the anonymous account by applying a deny write access control entry (ACE) for the Web Anonymous Users and Web Applications groups. It also adds a "deny execute" ACL on command-line tools.

- Use separate accounts for separate applications. If the Web server hosts multiple applications, use a separate anonymous account for each application. Add the accounts to an anonymous Web users group, such as the Web Anonymous Users group created by IISLockdown, and then configure NTFS permissions using this group.

Secure or Remove Tools, Utilities, and SDKs

SDKs and resource kits should not be installed on a production Web server. Remove them if they are present.

Ensure that only the .NET Framework Redistributable package is installed on the server and no SDK utilities are installed. Do not install Visual Studio .NET on production servers.

Ensure that access to powerful system tools and utilities, such as those contained in the \Program Files directory, is restricted. IISLockdown does this for the Administrator.

Debugging tools should not be available on the Web server. If production debugging is necessary, then the Administrator should create a CD that contains the necessary debugging tools.

Remove Sample Files

Sample applications are typically not configured with high degrees of security. It is possible that an attacker could exploit an inherent weakness in a sample application or in its configuration to attack the Web site. Remove sample applications to reduce the areas where the Web server can be attacked.

Additional Considerations

Also consider removing unnecessary Data Source Names (DSNs). These contain clear text connection details used by applications to connect to OLE DB data sources. Only those DSNs required by Web applications should be installed on the Web server.

Step 7. Shares

Remove any unused shares and harden the NTFS permissions on any essential shares. By default all users have full control on newly created file shares. Harden these default permissions to ensure that only authorized users can access files exposed by the share. In addition to explicit share permissions, use NTFS ACLs for files and folders exposed by the share.

During this step:
- Remove unnecessary shares.

Remove all unnecessary shares. To review shares and associated permissions, run the Computer Management MMC snap-in, and select **Shares** from **Shared Folders**.

- Restrict access to required shares.

  Remove the Everyone group and grant specific permissions instead. Everyone is used when the Administrator does not have restrictions on who should have access to the share.

Additional Considerations

If remote administration of the server is not allowed, remove unused administrative shares, for example **C$** and **Admin$**.

Note Some applications may require administrative shares. Examples include Microsoft Systems Management Server (SMS) and Microsoft Operations Manager (MOM).

Step 8. Ports

Services that run on the server use specific ports so that they can serve incoming requests. Close all unnecessary ports and perform regular audits to detect new ports in the listening state, which could indicate unauthorized access and a security compromise.

During this step:

- Restrict Internet-facing ports to TCP 80 and 443.
- Limit inbound traffic to port 80 for HTTP and port 443 for HTTPS (SSL).
- For outbound (Internet-facing) NICs, use IPSec or TCP filtering.
- Encrypt or restrict intranet traffic.

  For inside (intranet-facing) NICs, if there is no secure data center and there is sensitive information passing between computers, one might need to consider whether to encrypt the traffic and whether to restrict communications between the Web server and downstream servers (such as an application server or database server). Encrypting network traffic addresses the threat posed by network eavesdropping. If the risk is deemed sufficiently small one may choose not to encrypt the traffic.

  The type of encryption used also affects the types of threats that it addresses. For example, SSL is application-level encryption, whereas IPSec is transport layer encryption. As a result, SSL counters the threat of data tampering or information disclosure from another process on the same machine, particularly one running under a different account, in addition to the network eavesdropping threat.

Step 9. Registry

The registry is the repository for many vital server configuration settings. As such, one must ensure that only authorized administrators have access to it. If an attacker is able to edit the registry, he or she can reconfigure and compromise the security of the server.

During this step:

- Restrict remote administration of the registry.
- Secure the SAM (stand-alone servers only).

Restrict Remote Administration of the Registry

The Winreg key determines whether registry keys are available for remote access. By default, this key is configured to prevent users from remotely viewing most keys in the registry, and only highly privileged users can modify it. On Windows 2003, remote registry access is restricted by default to members of the Administrators and Backup Operators groups. Administrators have full control and backup operators have read-only access.

The associated permissions at the following registry location determine who can remotely access the registry.

> HKLM\SYSTEM\CurrentControlSet\Control\SecurePipeServers\winreg

To view the permissions for this registry key, run Regedt32.exe, navigate to the key, and choose **Permissions** from the **Security** menu.

Note   Some services require remote access to the registry.

Secure the SAM (Stand-Alone Servers Only)

Stand-alone servers store account names and one-way (non-reversible) password hashes (LMHash) in the local Security Account Manager (SAM) database. The SAM is part of the registry. Typically, only members of the Administrators group have access to the account information. Although the passwords stored in the SAM and password hashes are not reversible, if an attacker obtains a copy of the SAM database, the attacker can use brute force password techniques to obtain valid user names and passwords.

Restrict LMHash storage in the SAM by creating the key (not value) **NoLMHash** in the registry as follows:

> HKLM\System\CurrentControlSet\Control\LSA\NoLMHash

Step 10. Auditing and Logging

Auditing does not prevent system attacks, although it is an important aid in identifying intruders and attacks in progress, and can assist in diagnosing attack footprints. Enable auditing on the Web server and use NTFS permissions to protect the log files so that an attacker cannot conceal his actions by deleting or updating the log files in any way. Use IIS W3C Extended Log File Format Auditing.

During this step:
- Log all failed logon attempts.
- Log all failed actions across the file system.
- Relocate and secure the IIS log files.
- Archive log files for offline analysis.
- Audit access to the Metabase.bin file.

Log All Failed Logon Attempts

Log failed logon attempts to be able to detect and trace suspicious behavior.

To audit failed logon attempts:

1. Start the Local Security Policy tool from the Administrative Tools program group.

2. Expand **Local Policies** and then select **Audit Policy**.

3. Double-click **Audit account logon events**.

4. Click **Failure** and then **OK**.

Logon failures are recorded as events in the Windows security event log. The following event IDs are suspicious:

531. This means an attempt was made to log on using a disabled account.

529. This means an attempt was made to log on using an unknown user account or using a valid user account but with an invalid password. An unexpected increase in the number of these audit events might indicate an attempt to guess passwords.

Log All Failed Actions Across the File System

Use NTFS auditing on the file system to detect potentially malicious attempts. This is a two-step process.

To enable logging:

1. Start the **Local Security Policy** tool from the **Administrative Tools** program group.

2. Expand **Local Policies** and then select **Audit Policy**.

3. Double-click **Audit object access**.

4. Click **Failure** and then click **OK**.

To audit failed actions across the file system:

1. Start Windows Explorer and navigate to the root of the file system.

2. Right-click and then click **Properties**.

3. Click the **Security** tab.

4. Click **Advanced** and then click the **Auditing** tab.

5. Click **Add** and then enter Everyone in the **Name** field.

6. Click **OK** and then select all of the **Failed** check boxes to audit all failed events. By default, this applies to the current folder and all subfolders and files.

7. Click **OK** three times to close all open dialog boxes.

    Failed audit events are logged to the Windows security event log.

Relocate and Secure the IIS Log Files

By moving and renaming the IIS log files, it is much more difficult for an attacker to conceal his movements. The attacker must locate the log files before he or she can alter them. To make an attacker's task more difficult still, use NTFS permissions to secure the log files.

Move and rename the IIS log file directory to a different volume than the Web site. Do not use the system volume. Then, apply the following NTFS permissions to the log files folder and subfolders.

- Administrators: Full Control
- System: Full Control
- Backup Operators: Read

Archive Log Files for Offline Analysis

To facilitate the offline analysis of IIS log files, it is possible to use a script to

automate the secure removal of log files from an IIS server. Log files should be removed at least every 24 hours. An automated script can use FTP, SMTP, HTTP, or SMB to transfer log files from a server computer. However, if the Administrator enables one of these protocols, he or she should do so securely so as not to create new attack opportunities. Use an IPSec policy to secure ports and channels.

Audit Access to the Metabase.bin File

Audit all failures by the Everyone group to the IIS metabase.bin file located in \WINNT\System32\inetsrv\. Do the same for the \Metabase backup folder for the backup copies of the metabase.

Additional Considerations

Additionally, it is possible to configure IIS W3C Extended Log File Format Auditing. Select **W3C Extended Log File Format** on the **Web Site** tab of the Web site's properties dialog box. Then select **Extended Properties** such as URI Stem and URI Query.

Step 11. Sites and Virtual Directories

Relocate Web roots and virtual directories to a non-system partition to protect against directory traversal attacks. These attacks allow an attacker to execute operating system programs and utilities. It is not possible to traverse across drives. For example, this approach ensures that any future worm that allows an attacker to access system files will fail.

During this step:
- Move the Web site to a non-system volume.
- Disable the parent paths setting.
- Remove potentially dangerous virtual directories.
- Remove or secure RDS.
- Set Web permissions.
- Remove or secure FrontPage Server Extensions.

Step 12. Server Certificates

If the Web application supports HTTPS (SSL) over port 443, install a server certificate. This is required as part of the session negotiation process that occurs when a client establishes a secure HTTPS session.

A valid certificate provides secure authentication so that a client can trust the server it is communicating with and secure communication so that sensitive data remains confidential and tamperproof over the network.

During this step, the server certificate is validated.

Validate the Server Certificate

Check the following four items to confirm the validity of the Web server certificate:
- Check that the valid from and valid to dates are in range.
- Check that the certificate is being used correctly. If it was issued as a server certificate, it should not be used for email.
- Check that the public keys in the certificate chain are all valid up to a trusted root.

- Check that it has not been revoked. It must not be on a Certificate Revocation List (CRL) from the server that issued the certificate.

Snapshot of a Secure Web Server

A snapshot view that shows the attributes of a secure Web server allows for comparing settings with Acme's own Web server. The settings shown in Table H-3 are based on Web servers that host Web sites that have proven to be very resilient to attack and demonstrate sound security practices. By following the proceeding steps, the Administrator can generate an identically configured server, with regard to security.

Table H-3: Snapshot of a Secure Web Server

Component	Characteristics
Patches and Updates	Latest service packs and patches are applied for Windows, IIS, and the .NET Framework.
Services	Unnecessary services are disabled.
	NNTP, SMTP, and FTP are disabled unless they are required.
	WebDAV is disabled or secured if used.
	Service accounts run with least privilege.
	ASP.NET Session State service is disabled if not required.
Protocols	The NetBIOS and SMB protocols are not enabled on the server.
	The TCP stack has been hardened.
Accounts	Unused accounts are removed.
	Guest account is disabled.
	The default administrator account is renamed and has a strong password.
	Default anonymous account (IUSR_Machine) is disabled.
	Custom anonymous account is used for anonymous access.
	Strong password policies are enforced.
	Remote logons are restricted.
	Null sessions (anonymous logons) are disabled.
	Approval is required for account delegation.
	Shared accounts are not used.Membership of local administrators group is restricted (ideally to two members).
	Administrators are required to log on interactively (or a secure remote administration solution is implemented).
Files and Directories	**Everyone** group has no rights to system, Web, or tools directories.
	Anonymous account has no access to Web site content

			directories and system utilities.
			Tools, utilities, and SDKs are removed or secured.
			Sample files are removed.
			Unnecessary DSNs are removed.
		Shares	Unused shares are removed from the server.
			Access to required shares is secured (shares are not enabled to "Everyone" unless necessary).
			Administration shares (C$ and Admin$) are removed if not required.
		Ports	All ports except 80 and 443 (SSL) are blocked, especially vulnerable ports 135–139 and 445.
		Registry	Remote administration of the registry is prevented.
			SAM has been secured (stand-alone servers only).
		Auditing and Logging	Login failures are logged.
			Object access failures by the **Everyone** group are logged.
			Log files are relocated from %systemroot%\system32\LogFiles and secured with ACLs: Administrators and System have full control.
			IIS logging is enabled.
			Log files are regularly archived for offline analysis.
			Access to the metabase.bin file is audited.
			IIS is configured for W3C Extended Log File Format Auditing.
		IIS	
		Sites and Virtual Directories	Web roots and virtual directories are located on separate volumes from the system volume.
			Parent Paths setting is disabled.
			Dangerous virtual directories are removed (IIS Samples, MSADC, IISHelp, Scripts, and IISAdmin).
			RDS is removed or secured.
			Web permissions restrict inappropriate access.
			Include directories restrict Read Web permissions.
			Folders with Anonymous access restrict Write and Execute Web permissions.
			Secured folders that allow content authoring allow Script Source Access Web permissions while all other folders do not.
			FPSE is removed if not required.
		Script Mappings	Unused script-mappings are mapped to 404.dll: .idq, .htw, .ida , .shtml , .shtm , .stm , idc, .htr , .printer.
			Note The 404.dll is installed when the Administrator runs

		the IIS Lockdown tool.
	ISAPI Filters	Unused ISAPI filters are removed.
	IIS Metabase	Access to IIS Metabase is restricted with NTFS permissions.
		Banner information is restricted; the content location in HTTP response headers is hidden.

Staying Secure

Monitor the security state of the server and update it regularly to help prevent newly discovered vulnerabilities from being exploited. To assist in keeping the server secure:

- Audit group membership.

  Keep track of user group membership, particularly for privileged groups such as Administrators. The following command lists the members of the **Administrators** group: **net localgroup administrators**.

- Monitor audit logs.

  Monitor audit logs regularly and analyze the log files by manually viewing them or use the technique described in the Microsoft Knowledge Base.

- Stay current with service packs and patches.

  Set up a schedule to analyze the server software and subscribe to security alerts. Use MBSA to regularly scan the server for missing patches.

- Perform security assessments.

  Use MBSA to regularly check for security vulnerabilities and to identify missing patches and updates. Schedule MBSA to run daily and analyze the results to take action as needed.

- Use security notification services.

  Use the Microsoft services to obtain security bulletins with notifications of possible system vulnerabilities.

  Additionally, subscribe to the industry security alert services shown. This allows the Administrator to assess the threat of a vulnerability where a patch is not yet available (e.g., CERT Advisory Mailing List, Windows Security Updates).

Other Implementation choices are as follows:

Setting a Minimum Password Length Method:
Click Start, then Run.
In the Run box type "gpedit.msc."
Under Computer Configuration click the + next to Windows Settings, then Security Settings, then Account Policy, then Password Policy.
Double-click Minimum password length and set a good sized password.

Logging Failed Login Attempts Method:
Click Start, then Run..
In the Run box type "gpedit.msc,"
Under Computer Configuration click the + next to Windows Settings, then

Security Settings, Local Policies, and click Audit Policy.
Double-click Audit account logon events, make sure success is checked, then check failure also.
Do the same for Audit logon events.
Now, any unsuccessful logins will be shown in the Security section of the Event Viewer. The following information about the log-in failure will be displayed:

Reason
User Name
Domain (or computer name if no domain is present)
Logon Type
Logon Process
Authentication Package
Workstation Name
Caller User Name
Caller Domain (or workgroup)
Caller Logon ID
Caller Process ID
Transited Services
Source Network Address
Source Port

If the Administrator notices this repeatedly from the same computer (it shows the workstation name and IP) then the Administrator can take appropriate actions.

Securing Security Options Method:
Click Start, then Run.
In the Run box type "gpedit.msc."
Under Computer Configuration click the + next to Windows Settings, then Security Settings, Local Policies, then click on Security options.

Disabling the Administrator account

Double-click Accounts: Administrator account status and set the radio button to Disabled.

Goal(s):	Prevent input validation attack.
Category:	Unauthorized attacks
Requirement(s):	UA-5) The system shall protect itself from input validation attack.
No.	AR-16
Misuse case	MC-16
Architectural Recommendation	Hide HTML source code
Implementation Choices	There is a variety of software applications that can assist in encrypting an HTML page and hiding the source code. Using JavaScript code is another mechanism that can be used in Hiding the HTML source code [scriptasylum 04]. It is not foolproof, but it does make it more difficult to read and understand the source code. Due to the nature of how these scripts work, the explanation may seem complicated. The Administrator doesn't *have* to know the ins and outs of these scripts, but it does help he/she understand how and why they work. Escape/Unescape The first section of the scriptasylum page explains how to "escape" any text, HTML, or Javascript to make it generally unreadable to the common user. URL escape codes are two-character hexadecimal (8-bit) values preceded by a % sign. They are used primarily in browser URLs and when making cookies for characters that otherwise would not work, usually because they are reserved characters such as spaces. For example, if there was an HTML filename of **page one**, the escaped URL code would look like **page%20one**. The **%20** is the escape code for a space. In general, the Administrator would only escape special characters (generally any character other than a-z, A-Z, and 0-9), but the script below escapes *all* the text by replacing all characters with their escaped equivalents. If one were to **fully** escape the words **page one**, they would appear as **%70%61%67%65%20%6F%6E%65**. The browser can handle escape codes, so they can be used without having to add any more script to decipher them. If the Administrator wants the browser to write escaped text to the page, enter the following: <script language="javascript"> document.write(unescape('%70%61%67%65520%6F%5E%65')); </script> Wrap the escape string in a set of quotes inside the built-in **unescape()** method, and then wrap that in a **document.write()** method. One could hide an email address using this message to prevent Web crawlers from acquiring it to use in spam emailings, yet visitors would still be able to read it. Encoding/Decoding Using the above method one could hide the entire HTML page, but there are two disadvantages: file size and the ease of "cracking" the code. When the page is

	fully escaped, every character becomes three characters. This triples the size of the page. The size is not a significant factor if the page is 1-10K bytes in size; but if it is a large page (>10K bytes), the file size increases rapidly. This would slow the load time for the dial-up connection surfers. Also, if someone were to look at the source code, it would be uncomplicated to decipher the code. The hacker can copy and paste the code and compose a script to display the normal content. There is no foolproof way (client-side) to prevent someone from viewing the source if they are determined; the best one can hope for is to make it inconvenient.

Goal(s):	The claimed identities of all users and client applications will be authenticated before they are allowed access.
	Protect from unauthorized attacks involving addition, modification, deletion, or replay of data in network.
Category:	Authentication
Requirement(s):	Refer to Requirement AN-1.
No.	AR-19
Misuse case	MC-01, MC-02, MC-03, MC-06, MC-08, MC-09, MC-10, MC-14
Architectural Recommendation	Implement strong role-based authentication control.
	Note: The newer version of ARCHIBUS may have more security features.
Implementation Choices	**Authenticating through IIS** [Microsoft 04e]**:**
	Authentication control can be accomplished though the IIS application by requiring users to confirm their identity by providing a valid Microsoft Windows name and password though these Web site authentication methods. This identification process is one of the many features of IIS; authentication can be set at the Web site, directory, or file level.
	The authentication methods that are set by default are **Anonymous access** and **Integrated Windows authentication**.
	Table H-4 Comparison of Web Site Authentication Methods

Method	How Passwords Are Sent	Crosses Proxy Servers and Firewalls	Client Requirements
Anonymous authentication	N/A	Yes	Any browser
Basic authentication	Base64 encoded clear text	Yes, but sending passwords across a proxy server or firewall in clear text is a security risk because Base64 encoded clear text is not encrypted.	Most browsers
Digest authentication	Hashed	Yes	IE ver. 5 or later
Advanced Digest authentication	Hashed	Yes	IE ver. 5 or later

Integrated Windows authentication	Hashed when NTLM is used; Kerberos ticket when Kerberos is used.	No, unless used over a PPTP connection	Internet Explorer 2.0 or later for NTLM; Windows 2000 or later with internet Explorer 5 or later for Kerberos
Certificate authentication	N/A	Yes, using SSL connection	IE and Netscape
.NET Passport authentication	Encrypted	Yes, using SSL connection	IE and Netscape

Note: If multiple authentication methods are configured, IIS attempts to negotiate the most restrictive method first, and then it works down the list of available authentication protocols.

<u>Authenticating through Sybase</u>

Sybase Central is a graphical management tool for Sybase products. It implements the Sybase enterprise management strategy, which calls for a single management console. Sybase Central connects to and manages Sybase products running on any Sybase-supported platform. Sybase Central for Adaptive Server Enterprise (known as the Adaptive Server Enterprise Plug-in) is bundled with the application and can be installed from any of the ASE CDs.

Authentication control can be accomplished though the Sybase application by using an ASE plug-in. Using a LDAP server or Microsoft Active Directory, many Sybase servers can authenticate users from a single administration and control point with no changes required to the client applications. With Sybase ASE's support for Pluggable Authentication Modules (PAM), corporate security systems can be directly integrated to validate user and administrative access.

Authentication can also be achieved by installing Security Component EP Security 2.5. Therefore AMS can make use of the following functionalities provided below [Sybase 01].
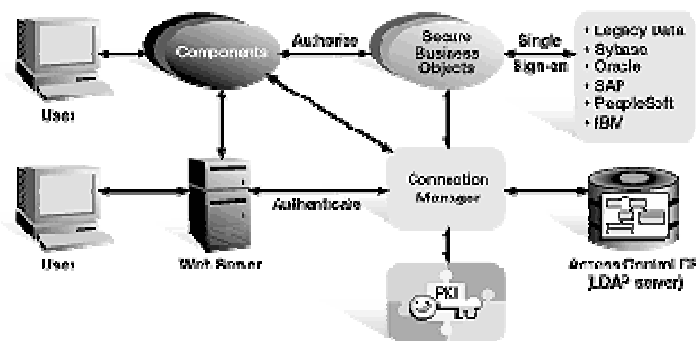
[Figure 1.] Sybase Enterprise Portal Security Framework

Sybase Enterprise Security Framework

The Sybase EP security framework is pictured in Figure 1. The center of this architecture is the Connection Manager and the Access Control Database, along with the set of Secure Business Objects. This provides the structure for user authentication and authorization across all of the portal applications and components, both Sybase and third party.

Connection Manager

The Connection Manager authenticates users to the portal, maintaining their security session state and interfacing with the Access Control Database and Secure Business Objects to perform access control decisions. It includes native support for username and password authentication as well as the use of digital certificates. The use of smart cards for storing and protecting digital certificates and private keys is supported and provides additional security.

The Connection Manager takes the authentication information provided by the user and validates it against the user's record in the Access Control Database. Regardless of the specific means employed to authenticate users, all of their sensitive communication with the portal is encrypted using the industry standard SSL. In addition, the Connection Manager includes an API that can be used to authenticate users through an external mechanism, such as RACF.

Upon successful login, the Connection Manager retrieves the user's profile from the Access Control Database and then creates and maintains an authentication string for the duration of the user's login session. This authentication string will be used by every user as the basis for authentication to other portal components, as well as access control decisions. The Connection Manager, along with the rest of the Sybase EP security framework, supports single sign-on across the portal.

Another way to authenticate within Sybase is to enable Certificate Authentication to EAServer 4.1.

Authenticating Through ARCHIBUS:

It is possible to personalize the ARCHIBUS/FM application by changing navigation, security, and control information without affecting the database structure. It is also possible to customize the way data is entered and extracted in accordance with Acme's business practices by taking advantage of technologies such as XML, ARCHIBUS/FM ActiveX objects, and Web templates. Database translation (Sybase, oracle and MS/SQL server) can be used for security configuration.

Custom Authentication

Custom authentication means creating one's own authentication mechanisms, such as ISAPI authentication filters, ASP pages, or Common Gateway Interface (CGI) applications.

Goal(s):	Ensure that essential services continue during and after an attack by
	• resisting attacks, especially with regard to essential services
	• recognizing attacks and their associated damage
	• recovering essential and full services after an attack
	• evolving to become more resistant to similar attacks in the future
	Ensure that failure under attack is graceful, resulting in a degraded mode of operation that still provides essential services.
Category:	Survivability
Requirement(s):	SU-1) The system shall continue to fulfill its mission in the presence of an attack (possibly in minim and safe mode).
No.	AR-21,AR-22
Misuse case	MC21
Architectural Recommendation	Invest in redundant IT hardware to ensure business continuity.
	Invest in redundant network capacity to avoid network downtime and system availability.
Implementation Choices	**Important Note:** One of the most important things regarding investment in redundant hardware is that Acme must identify and acquire the hardware first before attempting to ensure business continuity and avoid network downtime and system availability. This should be considered by the client before opting for any of the implementation choices.
	Window Server 2003 [Shinder 04b]
	Windows Server 2003 based ISA firewall/VPN servers can be configured for high availability by taking advantage of the Windows Server 2003 Network Load Balancing (NLB) service. The NLB service provides two major features that aid in increasing the availability of VPN connections for the VPN clients:
	Fail over
	Fail over allows other members of an ISA firewall/VPN server array to service connection requests from VPN clients when one of the servers becomes unavailable. All VPN servers in the array "listen" for VPN connections on the same IP address. When a VPN session is disconnected after a VPN array member goes offline, the connection is re-established to another array member using the same IP address. The VPN user does not need to reconfigure the VPN client software to automatically reestablish the connection.
	Load balancing for VPN connections
	VPN sessions can be processor intensive. Data encryption and decryption can take a significant percentage of the processor cycles available to the ISA firewall/VPN server per unit time. The NLB service can automatically split connections across all array members so that no single member of the array receives a disproportionate number of connection requests. NLB attempts to evenly spread the connection requests across all members of the NLB ISA firewall/VPN server array.

Configuring the Network Load Balancing service

One of the features to the NLB service included with Windows Server 2003 is the new Network Load Balancing Manager. The NLB Manager allows the Administrator to create, configure, and manage NLB arrays using an intuitive graphical interface.

Create the array after Windows Server 2003 software is installed on the machines that will be members of the ISA firewall/VPN server array, but before the Routing and Remote Access service is enabled with the ISA Server 2000 VPN wizard.

Perform all array management tasks from LOCALISAVPN1. Perform the following steps to create the Windows Server 2003 NLB arrays:

Click **Start**, point to **Administrative Tools**, and click on **Network Load Balancing Manager**.

> The **Network Load Balancing Manager** console opens. There are no NLB arrays configured by default. The Administrator will need to create an NLB array that allows all of the ISA firewall/VPN servers to listen on a single IP address on the external interface.

Click the **Cluster** menu and click the **New** command.

Fill in the following information in the **Cluster Parameters** dialog box:

> **IP address**
> This is the virtual IP address used by all of the members of the NLB array. The NLB Manager will automatically bind this address to the external interface of all the array members.

> **Subnet mask**
> This is the subnet mask for the virtual IP address

> **Full Internet name**
> This is the Fully Qualified Domain Name used to access the cluster IP address for command line remote administration. Enter a name here if the Administrator chooses to allow command line remote administration. This name must also be entered into the public DNS.

> **Cluster operation mode**
> The Windows Server 2003 NLB service can operate in either Unicast or Multicast mode. Choose multicast mode unless Acme has Cisco routers or switches on the same network segment as the external interface and those routers or switches do not support mapping unicast IP addresses to multicast MAC addresses. Please refer to the Windows Server 2003 Help for more information about NLB, unicast, and multicast modes.

> **Allow remote control**
> Put a checkmark in this checkbox if Acme wishes to allow command line remote control of the NLB array parameters. Do not allow command line remote control on the external interface array. Do not enable this checkbox.

> **Remote password**
> If remote command line administration were available, the Administrator would enter a password in this text box.

> **Confirm password**
> If remote command line administration were available, the Administrator

would confirm the password in this text box.

Click **Next**.

The Administrator can add more virtual IP addresses to the array in the Cluster IP Addresses dialog box. Click the **Add** button to add more VIPs. In this example there will not be additional VIPs. Click **Next**.

A default rule appears in the **Port Rules** dialog box. It is possible to create customized port rules that determine how connections are load balanced across all the servers in the array. Click on the default port rule, and then click the **Edit** button.

The details of the default port rule appear in the **Add/Edit Port Rule** dialog box. The default port rule includes the following parameters:

Cluster IP address
This entry determines what IP address this rule applies to. The default port rule applies to all addresses in the NLB array.

Port range
This entry determines what inbound ports the rule applies to. The default port rule applies to all inbound ports.

Protocols
The Administrator can have the rule apply to TCP, UDP, or both. The default port rule applies to both TCP and UDP protocols. Note that the Windows Server 2003 NLB port rules can only be applied to TCP and UDP protocols. The Administrator cannot apply port rules to other protocols such as ICMP.

Filtering mode
There are three filtering modes:

Multiple host
Specifies whether multiple hosts in the cluster handle network traffic for the associated port rule. The default port rule applies to all hosts in the array and the **Affinity** setting is set to "Single."

Single host
Specifies that network traffic for the associated port rule be handled by a single host in the cluster according to the specified handling priority. This filtering mode provides port-specific fault tolerance for the handling of network traffic.

Disable port range
Specifies whether all network traffic for the associated port rule will be blocked.

Click **Next** on the **Port Rules** page.

Type in the name of the machine running the NLB Manager application in the **Host** text box on the **Connect** page. The NLB Manager is on LOCALISAVPN1. Click the **Connect** button. There will be a list of interfaces on this machine in the Interface available for configuring a new cluster list. Click on the external interface of the ISA firewall/VPN array member. In this example, the external interface is named WAN (this is the name that appears in the **Network and Dial-up Connections** window; we have renamed the interfaces to make them more descriptive). Click **Next**.

The details of the NLB array member appear on the **Host Parameters** page.

Priority

Specifies a unique ID for each host.

IP address

This is the IP address on the external interface of the NLB array member for traffic not associated with the cluster (for example, Telnet access to a specific host within the cluster). Type the IP address in standard Internet dotted notation (for example, w.x.y.z). This IP address is used to individually address each host in the cluster and hence should be unique for each host.

Subnet mask

This is for the subnet mask for the IP address specified. Type the mask in standard Internet dotted notation (for example, 255.255.255.0).

Default state

Specifies the default host state of the Network Load Balancing cluster when Windows is started. Select the "Started" option if the host is to immediately join the cluster when Windows is started. Select the "Stopped" option if the host is to start without joining the cluster. Select the "Suspended" option if the host is to start without joining the cluster and instead enter a suspended state.

Retain suspended state after computer restarts

Specifies whether the host will remain suspended when Windows is restarted when the host was suspended prior to shutting down.

Click **Finish**.

The details of the NLB array configuration can be seen in the log entry pane in the bottom of the console window.

Next add a second machine to the array. Right-click the name of the array in the left pane of the Network Load Balancing Manager console and click the **Add Host to Cluster** command.

On the **Connect** page, type in the name of the computer to add to the array in the **Host** text box. In this example add LOCALISAVPN2 to NLB array. Select the external interface of this second array member in the Interface available for configuring the cluster list. Click **Next**.

Click **Finish**.

The details of the array configuration in the log entry pane can be seen at the bottom of the console. Double-click on the log entry with the description "Update 2 succeeded [double-click for details…]."

The log entry provides verbose details associated with that entry. Click **OK** and close the Network Load Balancing Manager console.

Goal(s):	Ensure secure communication.
	Prevent information gathering and network eavesdropping.
Category:	Encryption
Requirement(s):	EN-2) Secure technologies shall be used to provide secure communications channels.
No.	AR-25
Misuse case	MC 13, MC-19
Architectural Recommendation	Secure communication channels between servers.
	Secure communication channels between server and workstation.
Implementation Choices	Encrypting Data Transported Over a TCP/IP Network
	EFS (feature used in Microsoft Server 2003) only encrypts data when it is stored on disk. To encrypt data as it is transported over a TCP/IP network, two optional features are available; Internet Protocol security (IPSec) and PPTP encryption.
	Internet Protocol Security (IPSec) is a transport layer mechanism through which Acme can ensure the confidentiality and integrity of TCP/IP-based communications between computers. IPSec also supports machine-based authentication. These features make IPSec suitable for providing a secure communication channel between machines that is transparent to all applications.
	Internet Protocol Security (IPSec) can be used to secure the data sent between two computers, such as an application server and a database server. IPSec is transparent to applications because encryption, integrity, and authentication services are implemented at the transport level. Applications continue to communicate with one another in the normal manner using TCP and UDP ports.
	Using IPSec the Administrator can:
	Provide message confidentiality by encrypting all of the data sent between two computers.
	Provide message integrity between two computers (without encrypting data).
	Provide mutual authentication between two computers.
	Restrict which computers can communicate with one another. The administrator can restrict communication to specific IP protocols and TCP/UDP ports.
	Secure communication between server and workstation using W2K and MS 2003.
	Microsoft Corp. describes how to configure Microsoft Windows 2000 IPSec to help secure an internal corporate network server against network-based attacks from untrusted computers. Acme can significantly enhance the ability of a server to defend against such attacks by requiring IPSec-authenticated, signed, and encrypted communication between computers. The resource also describes the security threats to, and the benefits of using IPSec on, an internal corporate network server and uses a scenario to describe the process of IPSec policy design for an internal corporate network. Although the focus of the guidance is Windows 2000, it also provides information about IPSec functionality enhancements for Window Server 2003 [Microsoft 03e].

Goal(s)	Protect network from unauthorized attacks.
Category:	Unauthorized Attacks
Requirement(s)	UA-6) The system shall enforce the use of firewall technology.
No.	AR-26
Misuse case	MC-17, MC-18
Architectural Recommendation	Set up firewalls with filtering rules between servers and workstations.
Implementation Choices	Firewall and filtering through MS Server 2003 [Shinder 04a] *Packet Filtering* Packet filtering should be enabled on the ISA Server. If packet filtering is not enabled, all ports that are opened by applications and services on the ISA Server will be open. The goal is to close off all ports on the external interface unless given explicit permission, thereby adhering to the principle of least privilege. It is possible to allow IP Routing when packet filtering is enabled. Local Address Table hosts must use Network Address Translation to access the Internet. Packets are not directly routed from the Internet to the internal network; there is no need for concern regarding packets being directly routed into the internal network. IP Routing *does* allow non-TCP/UDP packets to move outbound from the internal network. Configure the ISA Server to enable IP Routing and to allow PPTP (which uses IP Protocol 47 GRE packets) and ICMP. Access control over the routing of the packets is not possible when the Administrator permits IP routing. Enabling IP Routing allows users to have access to outbound non-TCP/UDP protocols (as long as the packet filter is in place to support them). Use Protocol Rules and Publishing Rules to control outbound and inbound access. Enable filtering of IP Options and IP Fragments. Common exploits on the Internet today bypass firewall protection through the use of fragmented packets. By filtering out IP fragments, however, some multimedia applications may not work correctly. Test such applications to ensure that they work with fragment filtering enabled. IP Options filtering should always be enabled. This prevents source routed packet attacks. *Incoming Web Proxy Listeners* Create and enable incoming Web Proxy listeners only if Acme plans to use Web Publishing Rules to publish Web Servers on the internal network. Otherwise eliminate all Web Proxy service listeners. Remove Incoming Web Proxy Listeners by selecting the Configure listeners individually per IP address. Delete any listeners that might be there. *Site and Content Rules* The default settings on an ISA Server, a default Site and Content Rule, permit wide-ranging access to all sites, all content, at all times, to everyone. Change this setting right away by either deleting the rule or permitting Domain Users. If Acme does not change this setting, an anonymous access rule will be in place, and Web Proxy clients will access the anonymous access rule. Anonymous access rules are applied before other rules. If Acme has an

anonymous access rule that *Denies* access, then the "Deny anonymous access" rule will be applied before the "Allow anonymous access" rule.

By implementing Enterprise Policies a default Site and Content is not created.

Protocol Rules

After the ISA Server is installed, Protocol Rules are required for outbound access; no outbound access is possible until the Administrator creates them.

After Acme determines the required protocols and who needs to use them, create the appropriate Protocol Rules.

Acme should make a formal policy to review the firewall, packet filter, and Web proxy logs periodically.

Publishing Rules

Publishing Rules allow Internet and other external network users to access services on the internal network. Protocol Rules are limited in their function of providing access control or protection from external network intrusion. By creating a Protocol Rule that allows access to the internal network servers, it is possible for anyone who has access to interact with the server service just as if they were on the internal network accessing the same service.

Web Publishing Rules permit control access based on user/group and client address sets. Server Publishing Rules allow Acme to grant access to the rule based on client address set. Once the user is authorized, he/she has the access rights of those granted to a legitimate user on the published server.

Alerts

The ISA Server will post an alert to the event logs in the event that an enabled alert was triggered. Acme has the option to create new alerts, but one cannot create any new alerts other than those that have already been configured on the server.

Acme should review all the alerts available in the Alerts node. For intrusion alerts the Administrator should configure an email to be sent to him or herself notifying him/her that the alert took place. Depending on the security environment, the Administrator may decide if he/she wants to stop particular services in the event of a major alert or start monitoring the system or a particular application.

If the default configuration is unaltered, all enabled alerts will report to the event logs.

Logging

Logging should always be enabled for each service. This is the default setting. Acme should *never* disable logging for any of the services because the logs are Acme's first line of defense against a major attack or troubleshooting a problem with access and access controls.

New log files should be created each day, and the log files should be copied from the ISA Server to a safe location each day so that they are available if an intruder or hardware failure makes it impossible to retrieve them. The format isn't important, unless Acme wants to use local time. Save the logs in **ISA Server file format**.

The more fields are logged, the more system resources will be required to log them. Review the fields included in the log files.

The ISA Server reports are constructed using the Log Summaries. The Administrator must select the **Enable Reports** option to create the reports. Enable the **Enable daily and monthly summaries** options.

Application Filters

The ISA Server comes prepackaged with multiple application filters. Several filters are important in generating security alerts. Those filters are

- the DNS intrusion detection filter
- the POP intrusion detection filter
- the SMTP filter

These application layer filters are able to examine the application layer data portion of the packet and alert the Administrator to certain security-related concerns about that data. Acme should enable each of these filters and enable each option for the DNS filter. The SMTP filter allows for a greater number of options when used together with the SMTP Message screener.

If the Administrator does not use the SMTP Message Screener feature, the SMTP filter will examine SMTP commands and look for buffer overflow conditions for these commands. The SMTP filter is disabled by default, but should be enabled before connecting the ISA Server to the Internet.

Unless the ISA Server is deployed in a hybrid environment, there is not much justification for using a SOCKS filter. In a Windows environment, all clients will have the firewall client software installed and thus will not need the SOCKS filter. The SOCKS filter can be used by dangerous applications, such as instant messengers.

LAT/LDT

The Local Address Table (LAT) defines what addresses are trusted by the ISA Server. Connections to trusted addresses are not handled by the ISA Server firewall service. This prevents consumption of processor cycles for internal client access to internal resources. One particular configuration dilemma is to try and "loop back" access to internal network resources through the ISA Server. The loopback situation is a result of a misconfigured DNS infrastructure and not a problem with the ISA Server itself.

Do not include external network addresses in the LAT. It is possible to expose the security configuration of the ISA Server if the Administrator enters external addresses in the LAT. The ISA Server will consider those addresses as local and will not subject requests from those clients to the rules engine.

The local domain table serves the same purpose as the LAT. Acme should put only local domains in the LDT. By placing external domains in the LDT, access to external resources in those domains will not be subject to the ISA Server rules engine. Any time an external domain is included in the LDT, the security configuration on the ISA Server is weakened.

VPN Settings

ISA Server includes wizards that walk the Administrator through the process of configuring a VPN server and a gateway-to-gateway VPN configuration. ISA Server can also act as a packet filter to only allow PPTP and L2TP/IPSec connections. All other VPN related configurations are made in the RRAS console.

If VPN connections are allowed, the Administrator should configure RRAS Policies to limit who and how VPN connections are made to the server.

If PPTP is used, the Administrator must require complex passwords. The level of security offered by PPTP is dependent on the complexity of the password. If the users use simple passwords, it will be relatively simple for an intruder to access the network via the VPN connection. If at all possible, the Administrator should use L2TP/IPSec as the VPN protocol. L2TP/IPSec requires the use of machine certificates. It can be difficult for an intruder to get access to a machine certificate from the organization. This provides an extra layer of security to the VPN solution.

Conclusion

The ISA Server is only one facet of the network security scheme. Host based security must also be implemented. Here are some other things one should think about in terms of securing the network:

- Control application access at the desktop via Group Policy.
- Perform regular computer audits for unapproved applications.
- Have network usage policies and strong punishment for violating those policies.
- Harden all machines, especially servers accessible to Internet users.
- Use IPSec for all internal network communications.
- Require smart cards or biometric access for authenticated access.

These are just a few of things one should consider when securing the network. No firewall product can do it all.

Now for the checklist:

— Do not install services and applications on the ISA Server.

— Harden the Windows 2000 OS.

— Always install the latest security updates to Win2k and ISA Server.

— Disable all services that aren't required by the base operating system and ISA Server.

— Do not install ISA Server on a domain controller unless it's a dedicated ISA Server domain and forest.

— Disable File and Printer sharing on the external interface.

— Disable Client for Microsoft Networks on the external interface.

— Disable NetBIOS over TCP/IP on the external interface.

— Change the method for resolving unqualified names by choosing the Append these DNS suffixes (in order) option in the DNS tab in the Advanced TCP/IP settings Properties dialog box.

— Determine whether the network infrastructure requires enabling the Microsoft Client, File and Printer sharing, and NetBIOS on the internal interface. If Acme does not require these features, try turning them off.

— Turn on packet filtering.

— Do not enable IP Routing unless absolutely required.

— Enable fragment filtering.

	— Enable intrusion detection.
	— Enable filtering of IP Options.
	— Remove all incoming Web Proxy listeners if Acme does not plan to use Web Publishing Rules.
	— Change the default anonymous access Site and Content rule so that it applies to domain users, or delete the rule entirely.
	— Use the principle of least privilege.
	— Create Protocol Rules only for required protocols.
	— Limit access to protocols to users that require them.
	— Allow access to Publishing Rules only for those that require access to the published server.
	— Configure the published server to allow access only to those that require access to the server.
	— Harden the published server as if the server were directly connected to the Internet.
	— Review the available alerts.
	— Configure important alerts with response actions as determined by the corporate security policies.
	— Store logs and summaries on a dedicated, extendable disk.
	— Increase the number of saved log files.
	— Copy the log files each day to a safe location.
	— Increase the number of saved summaries.
	— Enable the DNS, POP, and SMTP application filters.
	— Use the SMTP Message Screener if the Administrator requires detection of more than SMTP command buffer overflows.
	— Disable the SOCKS filter.
	— Put only internal network addresses in the LAT.
	— Put only internal network domains in the LDT.
	— Do not "loop back" access to internal network resources through the ISA Server.
	— Use RRAS Policies to control access to the VPN server.
	— Require complex passwords, especially if PPTP is used.
	— Migrate to an L2TP/IPSec VPN solution as soon as possible.

Goal(s):	Protect from unauthorized attacks
Category:	Unauthorized Attacks
Requirement(s):	UA-7) A form of intrusion detection shall be implemented in the system. If the system detects any corruption of data or messages, then the system shall record the security event; notify the system administrator in a timely manner.
No.	AR-27
Misuse case	MC21, MC-18
Architectural Recommendation	Set up an intrusion detection system.
Implementation Choices	An article titled "Setting Up an Intrusion Detection System" shows ways of implementing IDS system and the importance of an IDS system [Franklin 04].

Knowing an attacker's moves is crucial for minimizing damage to the network. An intrusion detection system (IDS) can help the administrator understand how the attacker is reaching the organization's system, how the system is responding, and how a successful breach may have tricked the system into launching new attacks.

Intrusion Detection - Implementation Steps [Schorr 04]
If intrusion detection is enabled, the firewall administrator can configure the following IP Packet intrusion trigger alerts:

- Windows out-of-band (WinNuke)
- land
- ping of death
- IP half scan
- UDP bomb
- port scan

Also available are DNS application filters that analyze all incoming traffic for specific intrusions against the corresponding servers. The DNS intrusion detection filters helps the Administrator to intercept and analyze DNS traffic destined for the internal network:

- DNS Hostname Length Overflow
- DNS Length Field Overflow
- DNS Zone Transfer from Privileged TCP/IP Ports (1-1024)
- DNS Zone Transfer from High TCP/IP Ports (above 1024)

The POP buffer overflow attack intrusion detection filter, when enabled, intercepts and analyzes POP traffic destined for the internal network.

To configure intrusion detection for IP Packet Filters:

1. In the console tree of **ISA Management**, click
   - Internet Security and Acceleration Server 2000
   - Arrays
   - Access Policy
   - IP Packet Filters

2. In the right-side pane, click **Configure Packet Filtering and Intrusion** |

Detection.

3. On the **General** tab, click **Enable packet filtering** and **Enable intrusion detection**.

4. On the **Intrusion detection** tab, click which of the following types of attacks should generate events:

    -Windows out-of-band (WinNuke)

    -land

    -ping of death

    -IP half scan

    -UDP bomb

    -port scan

5. If the Administrator selects **Port scan**, then do the following:

    -In **Detect after attacks on**, type the maximum number of well-known ports that can be scanned before generating an event.

    -In **Detect after attacks on**, type the total number of ports that can be scanned before generating an alert.

6. Click **OK** to save changes.

7. A dialog screen will appear asking if the Administrator wants to save the changes and give him/her the choice of restarting the Services immediately or later. Click on the choice and then click **OK**.

To configure intrusion detection for DNS and POP Application Filters:

1. In the console tree of ISA Management, click

    - Internet Security and Acceleration Server 2000

    - Arrays

    - Extensions

    - Application Filters

2. In the right pane, double-click **DNS Intrusion Detection Filter**.

    - Click on Enable on the general tab.

    - Click on the filters you wish to enable and click **OK**.

3. Double-click the POP intrusion detection filter and click the box to enable the filter. Click **OK**.

Goal(s):	Ensure secure communication.
	Prevent information gathering and network eavesdropping.
Category:	Encryption
Requirement(s):	EN-3) A secure communication channel shall be used to secure Web data transfer.
No.	AR-32
Misuse case	MC-12, MC-13, MC-19
Architectural Recommendation	Use HTTPS for server-to-client Web data transfer encryption.
Implementation Choices	**Mechanism using IIS and Microsoft Server 2003** HTTPS is a secure communications channel that is used to exchange information between a client computer and a server. It uses Secure Sockets Layer (SSL). The following is from a Microsoft.com article that describes how to configure the SSL/HTTPS service in Internet Information Services (IIS) and shows the Administrator how to configure Web server for SSL [Microsoft 04f]. **Configure Folder or Web Site to Use SSL/HTTPS** This procedure assumes that the site already has a certificate assigned to it. 1. Log on to the Web server computer as an administrator. 2. Click **Start**, point to **Settings**, and then click **Control Panel**. 3. Double-click **Administrative Tools**, and then double-click **Internet Services Manager**. 4. Select the Web site from the list of different served sites in the left pane. 5. Right-click the Web site, folder, or file for which the Administrator wants to configure SSL communication, and then click **Properties**. 6. Click the **Directory Security** tab. 7. Click **Edit**. 8. Click **Require secure-channel (SSL)** if the Administrator wants the Web site, folder, or file to require SSL communications. 9. Click **Require 128-bit encryption** to configure 128-bit (instead of 40-bit) encryption support. 10. To allow users to connect without supplying their own certificate, click **Ignore client certificates**. Alternatively, to allow a user to supply their own certificate, use **Accept client certificates**. 11. To configure client mapping, click **Enable client certificate mapping**, and then click **Edit** to map client certificates to users. Note: If the Administrator configures this functionality, he/she can map client certificates to individual users in Active Directory. The Administrator can use this functionality to automatically identify users according to the certificate they supply when they access the Web site. The Administrator can map users to certificates on a one-to-one basis (one certificate identifies one user) or he/she can map many certificates to one user. (A list of certificates is matched against a specific user according to specific rules. The first valid match becomes the mapping.)

	Secure Sockets Layer (SSL) provides a secure, encrypted connection between server and the client computer. SSL can protect private information when users connect across a public network such as the Internet. SSL support requires an SSL certificate, and this certificate has to be installed on the computer that is running Windows Server 2003. SSL must also be supported by the client software.

Goal(s):	Prevent input validation attack.
Category:	Unauthorized Attacks
Requirement(s):	Refer to Requirement UA-5.
No.	AR-33
Misuse case	MC-16
Architectural Recommendation	Use least privileged account to access the database.
Implementation Choices	**Least privilege mechanism through IIS** IIS 6.0 now helps protect against the most common method of attacks on Web servers. Using the principle of least privilege, Administrators should use an account with restrictive permissions to perform routine, non-administrative tasks and use an account with broader permissions only when performing specific administrative tasks. To accomplish this without logging off and back on, log on with a regular user account use the **Run as** command to run the tools that require the broader permissions. **Using Run as** **Run as** is a feature that provides users with a secondary logon capability. By using **Run as**, users can run applications or commands in a different security context without having to log off. **Run as** prompts the user for different credentials before running the application or command. Using Run as, the Administrator can open and run a program using a different account and security context than the one the Administrator is logged on with. So, the user can log on using a regular user account, then, using **Run as**, open an administrative program in the context of an administrative account. The administrative context is used only for that specific program and is available only until that program is closed. It is possible to use **Run as** through the user interface or as a command-line tool. More information and detail instructions can be found on the Microsoft Web site. **Note**: No information was found for least privilege access for the Sybase database.

Goal(s)	Prevent input validation attack.
Category:	Unauthorized Attack
Requirement(s):	Refer to Requirement UA-5.
No.	AR-34
Misuse case	MC-16
Architectural Recommendation	Use parameterized stored procedure for database access.
Implementation Choices	Managing and Monitoring Adaptive Server Enterprise Using Sybase Central:
	Managing Stored Procedures
	A **stored procedure** is a named collection of SQL statements and flow control statements. Once it is created, Administrators can use it repeatedly without the need to enter the SQL statements individually each time they want to repeat a procedure. This section describes the following:
	• creating a procedure
	• displaying procedure properties
	• updating user and group permissions on a stored procedure
	• creating a stored procedure
	A stored procedure that performs a select, execute, or data modification command must be owned by the same user as the object acted upon.
	<table><tr><td>Privileges</td><td>Only a database owner or a user or group with **create procedure** permission can create a stored procedure.</td></tr></table>
	To create a stored procedure:
	In the database hierarchy, select the Stored Procedures folder.
	From the File menu, choose New; then choose Procedure from the cascading menu. The **Create a New Procedure** wizard opens.
	Recompile
	Select the recompile option if the Administrator expects that the execution of the stored procedure may be different each time. For example, recompile if the data passed in its parameters changes so much that a query plan produced at execution would differ greatly from a plan that is stored.
	Deleting a Stored Procedure
	Before deletion of a stored procedure, be sure that no other objects reference it. If any objects reference it, edit those objects to avoid errors. To find out if other objects reference a stored procedure, check its dependencies. See "Displaying Stored Procedure Dependencies."
	Viewing Stored Procedure Code
	To edit a stored procedure:
	Select the stored procedure to edit.
	From the File menu, choose Open. The code editor opens with the code for the

stored procedure displayed.

The Administrator can edit the code, but he/she cannot execute the edited code against Adaptive Server. The value of editing the code would be to save it to a file or copy it to the clipboard to reuse in creating a new stored procedure.

Displaying Stored Procedure Properties

To display the properties of a stored procedure:

Select the icon for the stored procedure to display.

From the File menu, choose Properties.

Displaying Stored Procedure Dependencies

To display dependencies for a stored procedure:

Select the stored procedure whose dependencies are required to be displayed.

From the File menu, choose Dependencies.

From the Object Type drop-down list, select the type of object to view. The options for the Referenced By tab are stored procedures and triggers. The options for the References tab are stored procedures, extended stored procedures, tables, views, and user data types.

Setting Permissions for A Stored Procedure

Sybase Central lets allows for updating permissions for stored procedures as follows:

Grant and revoke **execute** permission on a stored procedure.

Grant permission to specific users, groups, or roles, or grant it using the **with grant** option so the recipient can also grant the permission to other users.

Revoke the permission from specific users, groups, or roles, or revoke it using the **with cascade** option to revoke it from the named user and all users who acquired permission from the named user, directly or indirectly.

To update stored procedure permissions:

Privileges	A stored procedure owner can grant and revoke **execute** permission on the stored procedure.

Display the stored procedure property sheet.

Click the Permissions tab.

In the Permissions For drop-down list, select Users, Groups, or Roles.

To change permissions for an object, click in the Exec column until it indicates the correct permission. The Legend at the bottom of the dialog box explains the symbols.

To see the properties of an object, select it in the list, then click the Properties button.

Click Apply.

Managing Extended Stored Procedures

Extended stored procedures allow for external functions from Adaptive Server. The external functions must be capable of calling a C language function and manipulating C language data types. Once the Administrator creates an extended stored procedure, use it as any stored procedure. Extended stored procedures are

contained in the Extended Stored Procedures folder.

Extended stored procedures can

- take input parameters
- return a status value indicating success or failure and the reason for failure
- return values of output parameters
- return result sets

On platforms that support dynamic link libraries (DLLs), the external functions are compiled into DLLs. On platforms that do not support DLLs, the external functions are compiled into a single shared library named *libxp*.

Creating An Extended Stored Procedure

Privileges	Only a System Administrator can create extended stored procedures.

To create an extended stored procedure:

Select the Extended Stored Procedures folder.

From the File menu, choose New. From the cascading menu, choose Extended Stored Procedure.

The **Create A New Extended Stored Procedure** wizard opens.

Deleting an Extended Stored Procedure

Before deleting an extended stored procedure, be sure that no other objects reference it. If any objects reference it, the Administrator must edit those objects to avoid errors. To find out if other objects reference a extended stored procedure:

Viewing Properties of an Extended Stored Procedure

To view or edit the properties of an extended stored procedure:

Select the extended stored procedure you want to edit.

From the File menu, choose Properties. The property sheet for the extended stored procedure opens.

Adding Parameters to an Extended Stored Procedure

The **Create an Extended Stored Procedure** wizard does not support inclusion of parameters for the extended stored procedure. If the Administrator wants to add parameters to an extended stored procedure, generate DDL for the extended stored procedure, then edit the DDL code. To apply the changes, execute the DDL using **isql** or SQL Advantage.

Setting Permissions for an Extended Stored Procedure

Set permissions for Extended Stored Procedures the same way permissions are set for stored procedures.

Extended Stored Procedure Dependencies

	The dependencies dialog box lists procedures or triggers that reference the extended stored procedure.
	Note: The Dependencies dialog box does not display objects the extended stored procedure references because this information is not stored in Adaptive Server.
	To view the dependencies of an extended stored procedure:
	Select the extended stored procedure.
	From the File menu, choose Dependencies.
	Configuring Adaptive Server to Use Extended Stored Procedures
	Extended stored procedures are run by an Open Server application called XP Server. When running an extended stored procedure, the performance of Adaptive Server can be affected. It is possible to set the Adaptive Server configuration parameters to control the effect of running extended stored procedures.

Goal(s):	Prevent input validation attack.
Category:	Unauthorized Attack
Requirement(s):	UA-8) The system shall protect itself from input validation attack by assuring file names are well formed.
No.	AR-35, AR-36
Misuse case	MC-16, MC22
Architectural Recommendation	Use regular expressions to perform through input validation and use regular expressions to make sure files names are well formed.
Implementation Choices	**<u>Regular Expressions</u>** [Microsoft 04g] Regular expressions provide a powerful, flexible, and efficient method for processing text. The extensive pattern-matching notation of regular expressions allows the Administrator to quickly parse large amounts of text to find specific character patterns; to extract, edit, replace, or delete text substrings. For many applications that deal with strings (such as HTML processing, log file parsing, and HTTP header parsing), regular expressions are an indispensable tool. The .NET Framework regular expression classes are part of the base class library and can be used with any language or tool that targets the common language runtime.

Policy Security Requirements

Goal(s)	Ensure that the system functions properly.
Category:	Survivability
Requirement(s)	SU-2) All installation must be approved and reviewed by managers. SU-3) Only System Administrators are permitted to install any software and/or hardware.
No.	PR-01, PR-12
Misuse case	MC 13, MC-15
Policy Recommendation	All installations must be approved and reviewed by managers.
	Only System Administrators are permitted to install any software and/or hardware
Implementation Choices	No "computer related" because "System admin" needed to implement solutions for this policy. However, software is available to detect any changes in the system.

Goal(s)	Ensure that authorized modifications (e.g., defect fixes, enhancements, and updates) do not accidentally defeat security mechanisms.
	Maintain the levels of security specified in the security requirements during user usage.
	Ensure that system maintenance does not unintentionally disrupt the security mechanisms of the application or system.
Category:	Survivability
Requirement(s)	SU-4) The operating system, applications, firewalls, and IDS must be patched routinely.
No.	PR-02, PR-08
Misuse case	MC-01, MC-03, MC-13, MC-15, MC-16, MC-17, MC-18, MC-19, MC-20 MC-21, MC-22
Policy Recommendation	The operating system, applications, firewalls, and IDS must be patched routinely.
Implementation Choices	Software patches provide solutions to known security issues. Check software provider Web sites periodically to see if there are new patches available for all software, applications, and tools used in the organization.
	In IIS 6.0 there's a feature called automated patch management.
	Part of the patch management in Windows Server 2003 operating systems, the new fault-tolerant architecture of IIS 6.0, means that the server does require shutdown in order to install hot fixes, including security hot fixes. The Administrator does not need to be logged on to the computer for the installation to occur. In addition, Auto Update version 1.0 provides three patch management options [Microsoft 03b]: • Notify of patch availability as soon as it's available. • Download the patch and notify of patch availability. • Scheduled install, which enables the patch to be downloaded and automatically installed at a time the Administrator chooses.

Goal(s):	Enforce audit mechanisms to detect unauthorized use and to support incident investigations.
	Ensure that the application or component collects, analyzes, and reports information about
	• all security-related events
	• the status (e.g., enabled vs. disabled, updated versions) of its security mechanisms
	• the use of its security mechanisms (e.g., access and modification by security personnel)
	Ensure that the application or component collects sufficient information regarding potential breaches of security to establish what events occurred, when they occurred, and who (or what) caused them.
	Enable security personnel to audit the status and usage of the security mechanisms.
Category:	Auditing
Requirement(s):	AU-2) Audit information must be reviewed routinely.
	AU-3) Log all incoming and outgoing traffic.
No.	PR-03, PR-10
Misuse case	MC-01, MC-02, MC-03, MC-04, MC-05, MC-06, MC-07, MC-08, MC-09, MC-10, MC-11, MC-12, MC-13, MC-16, MC-17, MC-18, MC-19, MC-22
Policy	Audit information must be reviewed routinely.
Recommendation	Log all incoming and outgoing traffic.
Implementation Choices	The following is from an article titled "Security audit action list for CIOs," which describes a variety of security audit options and how often these audits should take place [Tittel 03].
	Establishing a proper security posture is absolutely essential and involves well-known steps of risk assessment, threat analysis, and formulation of an organizational security policy. No tool does a better job of helping to maintain a proper security posture than a security audit. Security audits come in different strengths and each has its own appropriate uses and frequencies.
	Vulnerability scan
	The vulnerability scan is a type of security audit that systematically tests for vulnerability to specific well-known attacks, especially those based on failure to patch or update key software or infrastructure components and known points of access or attack. Vulnerability scans may be performed by in-house or out-of-house staff and should be conducted at least once a month, and immediately after potentially dangerous vulnerabilities are discovered or become well-known on the Internet.
	Security checklist review
	The security checklist review employs published or publicly available checklists for specific types of platforms, applications, or services to make sure that software is up to date, configurations locked down, and potential points of attack closed. At a minimum, such reviews should be conducted quarterly and immediately after

any new or upgraded installation is brought online.

Security policy review

A security policy review examines an organization's security policy in detail. For example, the Administrator might implement this type of review on software and devices, as described in employee documents, training, and legal agreements, as implemented in vendor and consulting relationships and agreements, and so forth, to check that current configurations, implementations, procedures, processes, and documentation agree with the security policy. Such reviews should be conducted at least yearly as part of a thorough external security audit. Whenever systems, processes, or procedures change, at least a partial policy review should be conducted for those parts of the policy that are (or might be) affected.

Physical security audit

Review physical access controls and emergency procedures for an organization's sites, buildings, server and equipment rooms, and any areas where proprietary assets are stored or used. This is particularly important for information systems and related assets because physical access to these items by the wrong person can lead to their theft or loss.

Annual external security audit

Routine or event-driven security audits may be conducted by in-house or out-of-house staff, but just as external auditors routinely perform annual financial audits, so also should external security auditors perform annual security audits. In both cases, such audits provide valuable insights into internal attitudes and practices, as well as feedback on policies, procedures, and guidelines that govern related systems.

Event-driven audits

By reviewing and assessing the existing security policy Acme can decide the type of security audits to be undertaken and at what frequency. These components can then be part of a regular security schedule with a frequency that varies from annual to monthly. Finally, the Administrator can also implement event-driven audits for security scanning as new vulnerabilities are discovered or for security checklist reviews as systems and platforms are updated.

Goal(s):	Develop and exercise contingency or disaster recovery procedures to ensure appropriate availability.
	Ensure that client's components and personnel are protected against destruction, damage, theft, or surreptitious replacement (e.g., due to vandalism, sabotage, or terrorism).
Category:	Disaster Recovery
Requirement(s):	DR-1) Develop disaster recovery and contingency plan.
	DR-2) Perform routine system and data backup.
No.	PR-05, PR-20
Misuse case	MC-01, MC-02, MC-03, MC-04, MC-05, MC-06, MC-07, MC-08, MC-09, MC-10, MC-14, MC-15, MC-16, MC-17, MC-18, MC 21, MC-22
Policy	Develop disaster recover contingency plan.
Recommendation	Perform routine system and data backup
Implementation Choices	According to an article titled "Contingency and Disaster Recovery Plan," the following are recommended steps.
	A Contingency and Disaster Recovery Plan Guide
	The first step in a contingency and disaster recovery planning project is to obtain support from the appropriate management to support the effort. The next step is to select a project team. Once these steps are in place, the project team should
	• perform a risk analysis
	• perform a business impact analysis
	• develop a plan from the results of the risk and business impact analysis
	• develop a testing program
	• develop a maintenance program
	• determine the recovery strategies
	• document the efforts
	• develop a contingency plan that is easy to understand and obtain
	• develop a contingency plan that addresses both immediate and long term needs
	• test the plan to be sure Acme can react appropriately and in a timely manner
	• implement disaster avoidance and prevention procedures
	• train the entire workforce in disaster recovery and contingency plans
	• review the plan periodically and retrain the workforce on a regular basis
	Work continuity during and following a disaster is the job of everyone, not just the project team. The entire workforce should be involved in some portion of disaster recovery and contingency plans.

Goal(s)	Ensure that the system functions properly.
Category:	Survivability
Requirement(s)	SU-6) Do not set up shared files/folders/drives on the network.
No.	PR-06
Misuse case	MC-08, MC-10, MC-11, MC-14
Policy Recommendation	Do not set up shares files/folders/drives on the network.
Implementation Choices	**Microsoft Recommendations:** **Shares** Remove all unnecessary file shares, including the default administration shares if they are not required. Secure any remaining shares with restricted NTFS permissions. Although shares may not be directly exposed to the Internet, having limited and secured shares reduces risk if a server is compromised. Detail Steps - Shares Remove any unused shares and harden the NTFS permissions on any essential shares. By default all users have full control on newly created file shares. Harden these default permissions to ensure that only authorized users can access files exposed by the share. In addition to explicit share permissions, use NTFS ACLs for files and folders exposed by the share. During this step: **Do not use shared accounts**. Do not create shared accounts for use by multiple individuals. Authorized individuals must have their own accounts. The activities of individuals can be audited separately and group membership and privileges appropriately assigned. Remove Unnecessary Shares Remove all unnecessary shares. To review shares and associated permissions, run the Computer Management MMC snap-in, and select **Shares** from **Shared Folders**. Restrict Access to Required Shares Remove the Everyone group and grant specific permissions instead. Everyone is used when the Administrator does not have restrictions on who should have access to the share. Additional Considerations If the Administrator does not allow remote administration of the server, remove unused administrative shares, such as **C$** and **Admin$**. **Note** Some applications may require administrative shares. Examples include Microsoft Systems Management Server (SMS) and Microsoft Operations Manager (MOM).

Goal(s)	Data and communications shall not be corrupted.
	Ensure that persons understand and have reasonable control over their private information, thereby minimizing potential bad press and loss of user confidence.
Category:	Privacy
Requirement(s)	PV-2) Enforce strong password policies.
	PV-3) Password-protect any necessary shared documentation.
	PV-4) Users should log out of AMS system or close browser as soon as their activities are done.
No.	PR-07, PR-13, PR-22
Misuse case	MC-01, MC-03, MC-06, MC-08, MC-09, MC-10, MC-11, MC-13, MC-14, MC-20
Policy Recommendation	Enforce strong password policies.
	Password-protect any necessary shared documentation.
	Users should log out of AMS system or close browser as soon as their activities are done.
Implementation Choices	Most authentication methods require the user to provide a password to prove their identity. These passwords are normally chosen by the user, who may want a simple password that is easily remembered. Strong passwords tend to be more difficult for an intruder to discern and, as a result, help provide an effective defense of the organization's resources. Passwords provide the first line of defense against unauthorized access to the organization.
	The Microsoft Windows Server 2003 has a feature that verifies the complexity of the password for the Administrator account during setup of the operating system. If the password is blank or does not meet complexity requirements, the **Windows Setup** dialog box appears, warning the Administrator of the dangers of not using a strong password, which is considered to be at least six characters long, does not contain all or part of the user's account name, and contains at least three of the four following categories of characters: uppercase characters, lowercase characters, base 10 digits, and symbols found on the keyboard (such as !, @, #).
	Password-cracking tools continue to improve, and the computers that are used to crack passwords are more powerful than ever. Password-cracking software uses one of three approaches: intelligent guessing, dictionary attacks, and brute-force automated attacks that try every possible combination of characters. Given enough time, the automated method can crack any password.
	An example of a strong password is *J*p2leO4>F*.
	It is important to educate users about the benefits of using strong passwords and to teach them how to create passwords that are actually strong. The Administrator can create passwords that contain characters from the extended ASCII character set.

Goal(s)	Prevent input validation attack.
Category:	Unauthorized Attack
Requirement(s)	UA-9) Follow the principle of lest privilege and use least privileged service account to run processes and access resources.
No.	PR-09
Misuse case	MC-08, MC-09, MC-11, MC-17, MC-20, MC-22
Policy Recommendation	Follow the principle of least privilege and use least privileged service account to run processes and access resources.
Implementation Choices	Using the principle of least privilege, Administrators should use an account with restrictive permissions to perform routine, non-administrative tasks and use an account with broader permissions only when performing specific administrative tasks. To accomplish this without logging off and back on, log on with a regular user account and use the **Run as** command to run the tools that require the broader permissions.

Goal(s)	Ensure that the system functions properly.
Category:	Survivability
Requirement(s)	SU-5) New systems on the network should be evaluated prior to deployment.
No.	PR-11
Misuse case	MC-17
Policy Recommendation	New systems on the network should be evaluated prior to deployment.
Implementation Choices	Evaluate expected new systems using the following suggestions: • Identify where the new system will be located in the network and which persons will deploy it. • Evaluate the technology components and asset management processes used to maintain the new system. • Identify who has access to the new system. • Determine how implementers evaluated the impact of the technology against current technical strategy, existing systems, and alternative approaches. • Ensure that sufficient policies, procedures, and guidelines are available to implementers to assist in proper implementation of new systems. • Identify and review equipment and component standards that ensure consideration of total cost, compatibility, and system integration. • Identify any support personnel or Help systems that enable implementers to achieve successful deployment. • Obtain technical strategy and objectives documentation for the new system. • Obtain and review any existing business case documents pertaining to the introduction of new technologies. • Determine how high-impact potential technologies are identified.

Goal(s):	Prevent input validation attack.
Category:	Unauthorized Attack
Requirement(s):	UA-10) Perform routine code review.
No.	PR-15
Misuse case	MC-16
Policy Recommendation	Perform routine code review.

Implementation Choices	The following is from "Security Code Review Guidelines" [Shostak 04], a guideline and checklist for security groups performing the code review and an attempt to provide development teams with information about what to look for in a review.
	
	Before programs may be placed in the firewall system, the source code should be reviewed for deficiencies in the areas of security, reliability, and operations.
	
	I. Documentation
	
	      The code must be accompanied by documentation. The documentation allows the review team to do a proper review and provides the support people with information that they will need to run the code in the firewall environment.
	
	    A. Architectural overview
	      This overview will include a diagram of the system being implemented and the place of the code under review in the system. The diagram should show the client, the proxy, and the server, all in relation to the firewall system. The functional overview must include information about what threats the code is expected to deal with and how it will deal with them. The use of cryptography for confidentiality, integrity, etc. should be outlined here.
	
	    B. Comments in code
	      We expect the code to have a reasonable number of comments. As a guide, each file should have a comment at the start, explaining what the code does, possibly a comment at the start of each function, and comments as needed to explain complex or obfuscated code. (Incidentally, a compilation of the per module header files might make a good basis for an overview document, although it will not be a complete overview.)
	
	II. Code (Security Issues)
	
	    A. Libraries
	      There are a number of security-related libraries that can provide some of the functions described below. () is in the process of reviewing these libraries, and may be able to make a recommendation.
	
	    B. Command line
	      The command line arguments of a program should be checked carefully, especially if the code is running with, or might be

invoked with, privileges.

    C.   Data checking

This check should be to see if the data is what expected (length, characters) is. Making a list of bad characters is not the way to go; the lists are rarely complete. A secure program should know what it expects, and reject other input.

    D.   System calls

All system or library calls must have their return values checked and errors handled. Not checking the results of a system or library call is unacceptable. The sole exception to this is that class of calls which are designed to either work or exit and thus cannot return failure. Certain library calls have historically been found to be associated with security problems because they do no checking and user input is often passed to them.

    E.   Atomicity

Many security holes are related to programmer expectations that the flow of their program is uninterrupted. File access should be atomic. Temporary files, if they exist, should be created with tmpfile().

III.   **Code (Reliability Issues)**

    A.   System calls should have their return status checked.

    B.   Signals should be caught and handled.

    C.   Configuration information should be in a configuration file, not hard coded into the program. See the section on configuration files.

    D.   Functions should perform bounds checking.

IV.   **Testing**

During the process of writing code, it should be tested for functionality and security. These tests should be made available to the review team, preferably in the form of a script. Tests should look for things such as buffer overflows, proper dataflow, resistance to unusual input (control characters, for example), off-by-one loop errors, possible unterminated loop situations, etc. The code must be run through available tools for code quality checking.

V.   **Miscellaneous**

    A.   Code size

The programs should be kept to a minimum of size and functionality, because all code, even when done reviewing it, has bugs. Acme should strive for the smallest code that is reasonable for the job and the code should also be kept simple and straightforward.

    B.   Code formatting

All code will be run through a standard formatter before review and printed with file names and line and page numbers on each page. Lines will be formatted to wrap at a reasonable length.

Goal(s)	Data and communications shall not be corrupted.
	Ensure that persons understand and have reasonable control over their private information, thereby minimizing potential bad press and loss of user confidence.
Category:	Privacy
Requirement(s)	PV-5) Users should not reveal their account names and passwords in any situation.
	PV-6) Require users to change their passwords periodically.
No.	PR-16, PR-24
Misuse case	MC-01, MC-03, MC-10, MC-20
Policy	Users should not reveal their account names and passwords in any situation.
Recommendation	Require users to change their passwords periodically.
Implementation Choices	The following are password protection standards recommended by The SANS Institute [SANS 04]:
	Do not use the same password for AMS accounts as for other non-AMS access (e.g., personal ISP account, option trading, benefits, etc.). Where possible, don't use the same password for various AMS access needs. For example, select one password for the Engineering systems and a separate password for IT systems. Also, select a separate password to be used for an NT account and a UNIX account. Do not share AMS passwords with anyone, including administrative assistants or secretaries. All passwords are to be treated as sensitive, Confidential AMS information.
	Here is a list of "don'ts":
	• Don't reveal a password over the phone to *anyone*.
	• Don't reveal a password in an email message.
	• Don't reveal a password to the boss.
	• Don't talk about a password in front of others.
	• Don't hint at the format of a password (e.g., "my family name").
	• Don't reveal a password on questionnaires or security forms.
	• Don't share a password with family members.
	• Don't reveal a password to co-workers while on vacation.
	If someone demands a password, refer them to the password policy or have them call someone in the Information Security Department.
	Do not use the "Remember Password" feature of applications (e.g., Eudora, Outlook, and Netscape Messenger).
	Again, do not write passwords down or store them anywhere in the office.
	Do not store passwords in a file on *any* computer system (including Palm Pilots or similar devices) without encryption.
	Change passwords at least once every six months (except system-level passwords that must be changed quarterly). The recommended change interval is every four months.
	If an account or password is suspected to have been compromised, report the incident to the System Administrator and change all passwords.

Goal(s)	Ensure that the application or component collects, analyzes, and reports information about
	• all security-related events
	• the status (e.g., enabled vs. disabled, updated versions) of its security mechanisms
	• the use of its security mechanisms (e.g., access and modification by security personnel)
	Enable security personnel to audit the status and usage of the security mechanisms.
Category:	Auditing
Requirement(s)	AU-4) Separate personnel review system administrator activities.
No.	PR-18
Misuse case	MC-02, MC-04, MC-05
Policy Recommendation	Separate personnel review sys admin's activities.
Implementation Choices	No "computer related" because "system admin" is needed to implement solutions for this policy.

Goal(s):	All users and client applications will be identified before they are allowed access.
	Protect from unauthorized attacks involving addition, modification, deletion, or replay of data in network.
Category:	Access Control
Requirement(s)	AC-4) Set clear and defined user access control for all users (Low, Medium, High, System Admin).
No.	PR-19
Misuse case	MC,01, MC-08, MC-10, MC-11, MC-20
Policy Recommendation	Set clear and defined user access control for all users (Low, Medium, High, System Admin).
Implementation Choices	**Best practices for permissions and user rights recommended by Microsoft [Microsoft 03c]:**
	Assign permissions to groups rather than to users
	Because it is inefficient to maintain user accounts directly, assigning permissions on a user basis should be the exception.
	Deny permissions should be used for certain special cases
	Use Deny permissions to exclude a subset of a group which has Allowed permissions.
	Use Deny to exclude one special permission when you have already granted full control to a user or group.
	If possible, avoid changing the default permission entries on file system objects, particularly on system folders and root folders
	Changing default permissions can cause unexpected access problems or reduce security.
	Never deny the Everyone group access to an object
	If the Administrator denies everyone permission to an object, that includes administrators. A better solution would be to remove the Everyone group, as long as the Administrator gives other users, groups, or computers permissions to that object.
	Assign permissions to an object as high on the tree as possible and then apply inheritance to propagate the security settings through the tree
	The Administrator can quickly and effectively apply access control settings to all children or a subtree of a parent object. By doing this, the Administrator gains a greater breadth of effect with the least effort. The permission settings established should be adequate for the majority of users, groups, and computers.
	Privileges can sometimes override permissions
	Privileges and permissions may disagree, and the Administrator should know what happens if they do.
	For permissions on Active Directory objects, make sure the Administrator understands the best practices specific to Active Directory objects
	Active Directory has its own set of best practices regarding permissions.

Goal(s):	Enforce audit mechanisms to detect unauthorized use and to support incident investigations. Ensure that the application or component collects, analyzes, and reports information about • all security-related events • the status (e.g., enabled vs. disabled, updated versions) of its security mechanisms • the use of its security mechanisms (e.g., access and modification by security personnel) Enable security personnel to audit the status and usage of the security mechanisms.
Category:	Auditing
Requirement(s):	AU-5) User activities must be periodically reviewed. AU-6) Configuration changes are stored and cross-reviewed. AU-7) A DBA and/or manager performs information integrity checks on a routine basis.
No.	PR-04, PR-14, PR-21
Misuse case	MC-01, MC02, MC03, MC-04, MC-05, MC-06, MC-07, MC-08, MC-09, MC-10, MC-14, MC-15, MC-16, MC-17, MC-18, MC-20, MC-22
Policy Recommendation	User activities must be periodically reviewed.
	Configuration changes are stored and cross- reviewed.
	A DBA and/or manager performs information integrity checks on a routine basis.
Implementation Choices	The following is from "Monitor and inspect system activities for unexpected behavior" [CERT/CC 01], a CERT article that describes in detail the importance of this policy and ways to implement it. System activities include those associated with system performance, processes, and users. Programs executing on a typical networked systems include a variety of operating system and network services, user-initiated programs, and special-purpose applications such as database services. Every program executing on a system is represented by one or more processes. Each process executes with specific privileges that govern what system resources, programs, and data files it can access and what it is permitted to do with them. The execution behavior of a process is represented by the operations it performs while running, the manner in which those operations execute, and the system resources it uses while executing. Operations include computations, transactions with files, devices, and other processes, and communications with processes on other systems via the network. User activities include login/logout, authentication and other identification transactions, the processes they execute, and the files they access. If the Administrator permits third party (vendor, contractor, supplier, partner, customer, etc.) access to Acme's systems and networks, the Administrator must monitor their access to ensure all their actions are authentic and authorized. This includes monitoring and inspecting their system activities.

Why this is important

The Administrator needs to verify that the systems are behaving as expected and that the processes executing on the systems are attributed only to authorized activities of users, administrators, and system functions.

Unexpected or anomalous system performance may indicate that an intruder is using the system covertly for unauthorized purposes. They may be attempting to attack other systems within (or external to) the network or they may be running network sniffer programs. A process that exhibits unexpected behavior may indicate that an intrusion has occurred. Intruders may have disrupted the execution of a program or service, causing it to fail or to operate in a way other than the user or administrator intended. For example, if intruders successfully disrupt the execution of access-control processes running on a firewall system, they may access the organization's internal network in ways that would normally be blocked by the firewall.

How to do it

Notify users that monitoring of process and user activities is being done.

Inform authorized users of the system about the scope and kinds of monitoring Acme will be doing and the consequences of unauthorized behavior. A common method for accomplishing this is to present a banner message immediately before user login. Without the presentation of a banner message or other warning, Acme cannot likely use log files and other collected data in any action the Administrator may choose to take against a user.

Review and investigate notifications from system-specific alert mechanisms (such as email, voice mail, or pager messages).

This includes notifications from
- users and other administrators via email or in person
- operating system alert mechanisms
- system management software traps
- intrusion detection systems
- custom alert mechanisms from service or application programs (including tools)

Review and investigate system error reports.

These types of notifications typically are produced by
- operating system error reporting mechanisms
- log file filtering tools
- vendor or custom-developed management software
- custom error reporting mechanisms from service or application programs (including tools)

Often an administrator will be able to configure error reporting at a number of criticality, severity, or priority levels when installing the system, service and application programs, and supporting tools.

Review system performance statistics and investigate anything that appears anomalous.

Statistics are generally produced by vendor or custom performance monitoring tools. Typical statistics include

- total resource use over time (CPU, memory [used, free], disk [used, free])

- status reported by systems and hardware devices such as print queues

- changes in system status, including shutdowns and restarts

- file system status (where mounted, free space by partition, open files, biggest file) over time and at specific times

- file system warnings (low freespace, too many open files, file exceeding allocated size)

- disk counters (input/output, queue lengths) over time and at specific times

- hardware availability (modems, network interface cards, memory)

- performance statistics meaningful for a specific server or host

- comparison of previous system performance statistics with current statistics

Unexpected shutdowns, reboots, and restarts can indicate the presence of a Trojan horse program that requires a shutdown or restart of a system or service.

Continuously monitor process activity (to the extent possible).

The examination of processes is complex, time consuming, and resource intensive. The degree to which the Administrator is able to identify suspicious processes depends on the knowledge of what processes he/she normally expect to be executing on a given system and how they should behave. Due to the large number of processes and their rapidly changing natures, it is impractical for the Administrator to monitor them continually him/herself. In addition, the amount and value of information that the Administrator can gather from a snapshot of currently executing processes may be very limited. This means that the Administrator must employ a variety of information-gathering and monitoring mechanisms to help collect and analyze data associated with processes and to alert the appropriate person to suspicious activity. One common approach with multi-user systems is to set up consoles (or separate terminal windows on workstations) that display the current status of processes and are updated at short intervals. Ideally, these consoles should be hard-wired to the systems for which they are displaying information. With strategic placement of these displays, the Administrator can take advantage of the experience of system administrators to notice unexpected activity that may not be picked up by more immediate alert mechanisms.

Identify any unexpected, unusual, or suspicious process behavior and the possible implications.

As a general guideline, the Administrator should look for

- missing processes

- extra processes

- unusual process behavior or resource utilization

- processes that have unusual user identification associated with them

Data from log files and other data collection mechanisms will help the Administrator to analyze the process behavior. These include the

- user executing the process

- process start-up time, arguments, file names

- process exit status, time duration, resources consumed

- amount of resources used (CPU, memory, disk, time) by specific processes over time; top "x" resource-consuming processes

- system and user processes and services executing at any given time
- means by which each process is normally initiated (administrator, other users, other programs or processes), with what authorization and privileges
- devices used by specific processes
- files currently open by specific processes

Look for
- processes running at unexpected times
- processes terminating prematurely
- processes consuming excessive resources (wall clock time, CPU time, memory, disk), which may be a sign of an impending denial-of-service condition or the use of a network sniffer
- unusual processes, such as password cracking, network packet sniffing or any other process not due to normal, authorized activities
- processes with unusually formatted output or arguments (for example, on UNIX systems, a process running as "./telnetd" instead of "/usr/sbin/telnetd")
- new, unexpected, or previously disabled processes or services. These can indicate that an intruder has installed his/her own version of a process or service or, for example, is running IRC services, Web services, FTP services, and so forth to allow them to distribute tools and files he/she has stolen (such as password files) to other compromised hosts.
- inactive user accounts that are spawning processes and using CPU resources
- a terminal exhibiting abnormal input/output behavior
- processes without a controlling terminal that are executing unusual programs
- an unusually large number of processes

Pay close attention to the processes associated with intrusion detection and other security tools. Intruders regularly compromise these tools to gain greater leverage and information and to generate decoy alerts to distract and waste the time of system administrators.

Identify any unexpected, unusual, or suspicious user behavior and the possible implications.

Data from log files and other data collection mechanisms will help the Administrator analyze user behavior. These include
- login/logout information (location, time): successful, failed attempts, attempted logins to privileged accounts
- login/logout information on remote access servers that appears in modem logs
- changes in user identity
- changes in authentication status, such as enabling privileges
- failed attempts to access restricted information (such as password files)
- keystroke monitoring logs
- violations of user quotas

Look for
- repeated failed login attempts including to privileged accounts
- logins from unusual locations or at unusual times including unusual or unauthorized attempts to login via a remote access server
- unusual attempts to change user identity

- unusual processes run by users
- unusual file accesses, including unauthorized attempts to access restricted files
- users logged in for an abnormal length of time (both short and long)
- a user executing an unexpected command
- a user working from an unusual terminal

If the Administrator notices unusual activity associated with particular users, initiate supplemental data collection mechanisms to gather detailed information about their activities. Many multi-user systems provide mechanisms to audit all processes associated with a particular user. Since process accounting logs tend to generate a great deal of information rapidly, the Administrator will need to allocate sufficient resources to store the data collected. Similarly, detailed network logging of all activity associated with all the systems accessed by a specific user can be voluminous, and the Administrator will need to allocate resources accordingly. Review the newly collected data often (at least daily) and rotate files regularly to minimize the amount of information the Administrator has to analyze at any given time.

Identify other unexpected, unusual, or suspicious behavior and the possible implications.

If the network interface card is in promiscuous mode, an intruder may be using this mode to run network sniffers for capturing passwords and other sensitive information. Refer to the **Other information** section at the end of this practice. However, keep in mind that legitimate network monitors and protocol analyzers will set a network interface in promiscuous mode as well. Doing some level of correlation analysis (determining when intrusion activity occurring in one part of the systems may be related to activity in another part) during the intrusion detection process will assist the Administrator in determining the full extent of any compromise and its characteristics. Logging information produced by vulnerability patches (updated software that corrects or closes a vulnerability), if provided by the vendor and if turned on, can aid in identifying a pattern where an intruder exploits more than one vulnerability before gaining access. For example, a failed logged attempt to probe for an old vulnerability (produced by the vulnerability patch) could be followed by a successful probe for a new vulnerability that is not logged. The presence of the vulnerability patch logging information along with other mechanisms such as integrity checking could alert the Administrator to this type of intruder action.

Periodically execute network mapping and scanning tools to understand what intruders who use such tools can learn about the networks and systems.

We recommend running mapping and scanning tools during non-business hours and when the Administrator is physically present because mapping tools can sometimes affect systems in unexpected ways. Eliminate or make invisible (if possible) any aspect of the network topology and system characteristics that Acme does not want to be known by intruders who use mapping tools.

Periodically execute vulnerability scanning tools on all systems to check for the presence of known vulnerabilities.

We recommend running such tools during non-business hours and when the Administrator is physically present because scanning tools can sometimes affect

systems in unexpected ways. Eliminate all vulnerabilities identified by these tools wherever possible. Many of these can be dealt with by updating configuration file settings and installing vendor-provided patches. Consider using scanning tools that include password analysis as part of the vulnerability assessment. Such analysis may include the identification of weak, non-existent, or otherwise flawed passwords such as those that can be determined using brute force or dictionary-based attacks.

If the Administrator is reviewing system activities on a host other than the one being monitored, ensure that the connection between them is secure.

Policy considerations:

Acme's networked systems security policy should
- require that users be notified that monitoring of process and user activities will be done and the objective of such monitoring
- specify the responsibilities and authority of designated systems administrators and security personnel to examine systems, processes, and user activity for unexpected behavior
- specify what forms of unexpected behavior users should watch for. Require users to report any such behavior to their designated security officials and system administrators.
- specify what software and data users and administrators are permitted to install, collect, and use, with explicit procedures and conditions for doing so
- specify what programs users and administrators are permitted to execute and under which conditions

Other information

One common activity of intruders is to gather information from the traffic on Acme's networks to find user account names, passwords, and other information that may facilitate their ability to gain access to Acme's systems. They do this by breaking into one system on the network and installing and executing a sniffer program. This program collects information about connections established between systems from network data packets as they arrive at or pass by the compromised system. To hide this illicit activity on compromised systems, intruders typically modify log files and replace programs that would reveal the presence of the sniffer program with Trojan horse versions. The substitute programs appear to perform the same functions but exclude information associated with the intruders and their activities. In many documented cases of this type of intrusion, the intruders' activities went unnoticed for a considerable amount of time, during which they collected enough information to gain privileged access to several other systems.

This underscores the importance of using verified software to examine Acme's systems and the need to verify the integrity of the files. Unfortunately, there are several sophisticated collections of programs that intruders can use to rapidly gain access to systems and "set up shop" to install and execute a sniffer. This means that the only method an Administrator may have to catch such activity is to use verified software to examine processes on the systems for unexpected behavior. Processes associated with a sniffer will typically have transactions with a network interface that has been placed in promiscuous mode, as well as a file or network connection to which the information gathered from network packets is being sent.

Goal(s):	All users and client applications will be identified before they are allowed access.
	Protect from unauthorized attacks involving addition, modification, deletion, or replay of data in the network.
Category:	Access Control
Requirement(s):	AC-5) Users should not have rights or access levels beyond those which are prescribed by their job responsibilities.
No.	PR-23
Misuse case	MC-01, MC-02, MC03, MC-06, MC-08, MC-09, MC-10
Policy Recommendation	Users should not have rights or access levels beyond those which are prescribed by their job responsibilities.
Implementation Choices	This policy recommendation can be achieved by implementing some of these recommendations from Microsoft [Microsoft 03d].
	Windows Server 2003 introduces an authorization interface called Authorization Manager, which includes role-based access control. Authorization Manager provides a framework for business process applications that require representing the organizational model within the application security framework.
	Role-Based Access Control
	In contrast, role-based access control is a user-centric authorization model. Rather than enumerating objects in the system for each user and assigning privileges, role-based access control allows administrators to specify access control in terms of the organizational structure of a company. Role-based access control provides a central object—a role—that a user is assigned to perform a particular job function. A role directly implies authorization permissions on some defined set of resources.
	With role-based access control, permissions are granted not through low-level rights, but rather through higher level abstractions corresponding to application operations and tasks. Operations run as a single unit, whereas tasks may be composed of multiple operations (and other tasks). Consider an example application that allows users to report project status, publish status for viewing, and view status. Status is reported, published, and viewed in a Web-based interface. When published, the database is updated and an email message is sent to interested parties.
	In the role-based access control model, the role is the interface an administrator uses to manage permissions and assignments. For example, a company can create a role called "User" that is defined in terms of the permissions users need for their jobs. Each user hired is assigned to the User role and instantly has all required permissions for that job. Similarly, users who leave the position are removed from the User role and no longer have User access. Since the role grants access in terms of a company's organizational model, it is more intuitive and natural for administrators to specify access control. Whereas ACLs work well for well-defined, persistent resources, the role-based model lends itself well to protecting workflow or groups of multiple distinct operations (for example, "read from database," "send email") to be performed by the application.
	After a role is defined, managing it is easy. The more difficult task is defining the role and the specifying access criteria in the first place. However, experience

shows that changing role access specification is rare once roles are defined; it is more common to change membership in a role. The guiding principle is to keep the activities that are easy common and to keep activities that are difficult rare.

Role-based access control in Windows Server 2003 also allows users to be collected into groups. Role-based access control groups are similar to groups in the Active Directory service, but they are maintained for a specific set of applications, a single application, or a scope within an application.

Authorization Manager introduces two types of application-scoped groups:

Application Basic Group: Similar to Windows NT groups, the application basic group contains a list of members. Unlike Windows NT groups, it also has an additional list for nonmembers. The nonmembers list allows for exceptions, so a large group can be used but a smaller group, or particular, user can be excluded.

Lightweight Directory Access Protocol Query Group: A group defined by a Lightweight Directory Access Protocol (LDAP) query against the attributes of a given Active Directory user's account. At the time of access, the LDAP query is run to determine if the user is a member of that group. This allows for flexible group membership that remains up to date with the user's Active Directory account object. For example, a Managers group could contain an LDAP query that includes all users who have direct reports.

The simplicity of Authorization Manager arises from its implementation of role-based access control. Authorization administrators design roles as collections of tasks supported by an application, then assigns users and groups to the role to grant them the ability to perform those tasks. Application developers use secured logical objects that make sense both in the context of the application and in the context of the security administration model, simplifying both application development and administration.

Application Development Process with Authorization Manager

The incorporation of role-based access control within an application follows a common course:

- At application development time, identify roles, implement operations, and roll the operations up into tasks.
- At install time, call the appropriate APIs to create an Authorization Store, create operations and tasks (and possibly some initial roles required by the application).
- At run time, the application initializes the Authorization Manager to connect to the Authorization Store and then establishes a connection to the section of the store specific to the application.

When a client connects to the application, it creates a client authorization context.

Implement custom behavior based on roles. Now available is the option to get the roles for that user and render a different UI based on their role (for example, a "manager" might see something different from a "consultant"). When an operation is performed, Access Check is called. The role that the user is in is enumerated; each task for each role is evaluated to see if the requested operation can be yielded from them.

Conclusion

Role-based access control available in Windows Server 2003 provides a simplified development model for line of business type applications. Both administrators and developers benefit from the natural framework that allows them to effectively model both organizational structure and business processes.

<u>Access Control through Sybase</u> [Sybase 03]

Access control can be accomplished though the Sybase application by using the ASE plug-in, which has security features that include role-based access control, proxy authorization, single sign-on, and a C2 certification. ASE includes a Policy-Based Access Control framework that provides a means of protecting data. Administrators can define security policies that are based on the value of individual data elements. The server then enforces these policies. Once a policy has been defined, it is invoked whenever the affected data is queried, whether through an application, ad hoc query, stored procedure, or view. This simplifies both the security administration of an ASE installation and the application development process, because it is the server, not the application, that enforces security. This allows developers to concentrate on implementing business functionality while administrators focus on defining a security policy to enforce consistently across the entire server. This is accomplished through the four combined capabilities of

- access rules
- the Application Context Facility
- login triggers
- domain integrity rules

Client's Own Authentication Mechanism

The client can also choose to use their own custom authentication mechanism to implement this architectural recommendation.

Appendix I   Architectural and Policy Recommendations – Categories

The Security Requirement Table guides the client in finding architectural and policy recommendations with ease. These recommendations address specific categories such as Access Control, Encryption, Auditing, Privacy, Authentication, Survivability, Disaster Control, and Unauthorized Attack.

Access Control
- AR-08
- PR-19
- PR-23

Encryption
- AR-12
- AR-25
- AR-32

Auditing
- AR-03
- PR-03
- PR-04
- PR-10
- PR-14
- PR-18
- PR-21

Privacy
- AR-10
- AR-17
- PR-07
- PR-13
- PR-16
- PR-24

Authentication
- AR-01
- AR-04
- AR-09
- AR-19

Survivability
- AR-21
- AR-22
- PR-01
- PR-02
- PR-08
- PR-11
- PR-12

Disaster Recovery
- PR-05
- PR-20

Unauthorized Attack
- AR-02
- AR-05
- AR-13
- AR-14
- AR-15
- AR-16
- AR-23
- AR-24
- AR-26
- AR-27
- AR-30
- AR-31
- AR-33
- AR-34
- AR-35
- AR-36
- PR-09
- PR-15
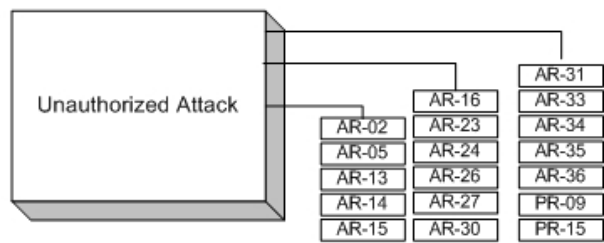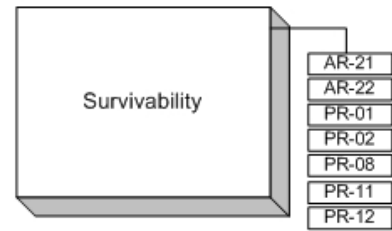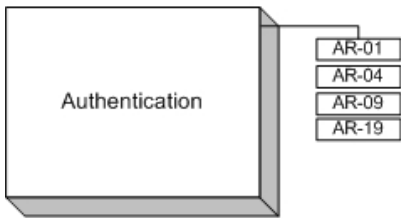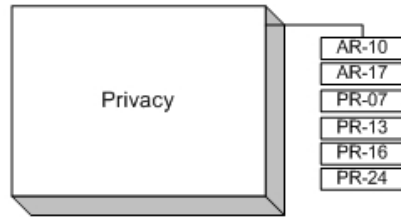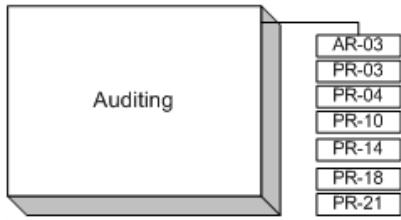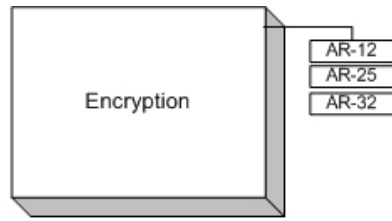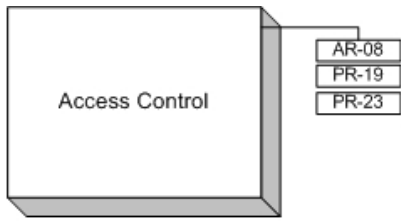
Note: AR-06, AR-07, AR-11, AR-18, AR-20, AR-28, AR-29 are missing because they are labeled as Medium Priority.

Appendix J Architectural and Policy Recommendations - Flow Diagrams

Flow diagrams provide the client with a tool for tracing requirements to implementation, thus enabling the client to know that every requirement has been implemented and no extraneous functionality has been added.

Goal:
Prevent input validation attack

UA-5
The system shall protect itself from input validation attack

→ AR-16, AR-33, AR-34,

↔ - Use Encoding Mechanism

UA-8
The system shall protect itself from input validation attack by assuring file names are well formed.

→ AR-35, AR-36

↔ -Using Sybase

UA-9
Follow the principle of least privilege and use least privileged service account to run processes and access resources.

→ PR-09

↔ -Principle of Lease Privilege

UA-10
Perform routine code review

→ PR-15

↔ - Security Code Guidelines

Goal:
Protect Network from unauthorized attacks.

UA-1
The system shall protect itself from viruses by using virus detection software with updated signatures. The virus detection software should also be run weekly.

→ AR-02 ↔ Virus Detection Using MS server 2003

UA-2
The system shall be able to determine when a buffer overflow attack has occurred. If attack takes place system should shut down and system administrator should be notified within a reasonable time.

→ AR-05 ↔ - Buffer Feature in IIS.
- Buffer Feature in Sybase .

UA-3
The system shall protect itself from malicious encoding mechanism.

→ AR-13, AR-14, AR-23, AR-24, AR-30, AR-31 ↔ Encode/Decode

UA-4
The system shall protect itself from attacks due to weak server configuration

→ AR-15 ↔ - Securing Web Server.
- Installation Recommendations.
- Other implementation choices using Microsoft Server 2003

UA-6
The system shall enforce the use of firewall technology

→ AR-26 ↔ - Firewall and filtering through MS Server 2003

UA-7
A form of intrusion detection shall be implemented in the system. If the system detects any corruption of data or messages, then the system shall record the security event; notify the system administrator in a timely manner.

→ AR-27 ↔ - Intrusion Detection Systems.
- Implementation Steps

Goal:

All users and client applications will be identified before they are allowed access.

Protect from unauthorized attacks involving addition, modification, deletion or replay of data in network

AC-4

Users should not have rights or access levels beyond those which are prescribed by his or her job responsibilities.

PR-23

-Role Based access Control
- Access control through Sybase
- Clients Authentication Mechanism

AC-4

Set clear and defined user access control for all users. (Low, Medium, High, System admin)

PR-19

- Permission and User Rights .

AC-2

The system shall not grant any client application access to one or more system services and data during a secure session without first identifying the client application.

AC-1

The system shall not grant any user access to one or more system services and data during a secure session without first identifying the user.

AR-08

-Access Control through IIS
- Access Control through Sybase
- Access Control through Archibus

AC-3

If no identification is provided, then the system shall record the security event; notify the operator within a reasonable time.

Goal:
The claimed identities of all users and client applications will be authenticated before they are allowed access.

Protect from unauthorized attacks involving addition, modification, deletion or replay of data in network

AN-1
Authentication control mechanism shall be enforced in production environment.

Authentication control will be done on user name and password or other user credentials

AR-01, AR-19

-Authenticating Through IIS
- Authenticating through Sybase
- Authenticating through Archibus

AN-2
Authentication control mechanism shall be enforced in production environment.

Authentication control will be done on the network level.

AR-04, AR-09

- Principle of Least privilege .

Goal:

Enforce audit mechanisms to detect unauthorized use and to support incident investigations.

Ensure that the application or component collects, analyzes, and reports information about:
O All security-related events.
O The status (e.g., enabled vs. disabled, updated versions) of its security mechanisms.
O The use of its security mechanisms (e.g., access and modification by security personnel).

Ensure that the application or component collects sufficient information regarding potential breaches of security to establish what events occurred, when they occurred, and who (or what) caused them.

Enable security personnel to audit the status and usage of the security mechanisms

AU-1
Separate personnel review Sys admin's activities.

PR-18

-No "computer related mechanism

AU-1
The system shall audit systems on network and user logging information. This shall be put into practice monthly

AR-03

-Using IMS Server 2003

AU-2
The system shall protect and review audit information.

PR-03, PR-10,

-Variety of Security Audits.
- Variety of Implementation.

AU-5
User activities must be periodically reviewed

AU-6
Configuration Changes are stored and cross- reviewed

AU-7
Perform information integrity checks on a routine basis by DBA and/or manager.

PR-04,PR-14, PR-21

- CERT's guidelines for inspecting and monitoring a system.

Goal:
Develop and exercise contingency or disaster recovery procedures to ensure appropriate availability.
Ensure that client's components and personnel are protected against destruction, damage, theft, or surreptitious replacement (e.g., due to vandalism, sabotage, or terrorism).

DR-1
Develop disaster recover contingency plan

DR-2
Perform routine system and data back-up.

PR-05, PR-20

- Contingency and Disaster Recovery Plan Guide

Goal:

Ensure the security and confidentiality of user records and information.

Protect against any anticipated threats or hazards to the security or integrity of user credentials.

Ensure that confidential communications and data are kept private.

EN-1
The system's data and communication shall be encrypted

AR-12

-Encryption through IIS
- Encryption through Sybase

Goal:

Ensure secure communication

Prevent Information gathering and Network Eavesdropping

HTTPS shall be used to secure web data transfer

EN-2
Secure technologies shall be used to provide secure communications channels

EN-3
HTTPS shall be used to secure web data transfer

AR-25

AR-32

- Encrypting Data Transported Over a TCP/IP Network
- Secure between server and workstation using W2K and MS 2003

- Mechanism using IIS and Microsoft Server 2003

Goal:

Data and communications shall not be corrupted.

Ensure that persons understand and have reasonable control over their private information. Minimizing potential bad press and loss of user confidence

PV-1
The system shall not permit any user data to be retrieved by an attacker.

AR-10, AR-17

-Implement using MS Server 2003

PV-2
Enforce Strong password policies.

PV-3
Password Protects any necessary shared documentation.

PR-07, PR-13

-Mechanism using MS Server 2003

PV-4
Users should not reveal their account names and passwords in any given situation

PV-5
Require user to change their passwords periodically

PR-16, PR-24

Password Protection Standards

Goal:
Ensure system functions properly

⇒

SU-2
All installation must be approved and reviewed by managers.

SU-3
Only System Administrators are permitted to install any software and/or hardware

⇒ PR-01, PR-12 ↔ -No computer related implementation.

SU-5
The system shall be evaluated.

⇒ PR-11 ↔ - Outlined a variety of suggestions.

Goal:
Ensure that authorized modifications (e.g., defect fixes, enhancements, and updates) do not accidentally defeat its security mechanisms.

Maintain the levels of security specified in the security requirements during user usage.
Ensure that system maintenance does not unintentionally disrupt the security mechanisms of the application, or system

Ensure system functions properly

⇒

SU-4
The system shall be protected during the replacement of a hardware or software component

⇒ PR-02, PR-08, ↔ -Using IIS 6.0

Goal:
Ensure that essential services continue during and after an attack by:
o Resisting attacks, especially with regard to essential services.
O Recognizing attacks and their associated damage.
O Recovering essential and full services after an attack.
O Evolving to become more resistant to similar attacks in the future.

Ensure that failure under attack is graceful, resulting in a degraded mode of operation that still provides essential services.

⇒

SU-1
The system shall continue to fulfill its mission in the presence of an attack (possibly in minim and safe mode)

⇒ AR-21, AR-22, ↔ -Using Window Server 2003 - Authenticating through Sybase - Authenticating through Archibus

Appendix K   Architectural Recommendation Costs

Legend:

D: Denial of service
P: System penetration
S: Sabotage of data
T: Theft of proprietary info
U: Unauthorized access by insiders
V: Virus
W: Active Wiretap/Network Eavesdropping

Terms:
Category of Threat: Set of related misuses and attacks that pose threat(s) to the organization.
Implementation Cost: Cost needed to implement (or configure) an architectural recommendation. Could include training costs associated with implementation. Usually a one-time fee expressed in dollars.
Maintenance Cost: Cost needed to maintain an architectural recommendation after implementation. Includes time spent on recommendation. It is expressed in dollars per year.
Software Cost: Cost of any software that needs to be purchased, installed, and/or configured in order to implement an architectural recommendation. Usually a one-time fee expressed in dollars.
Hardware Cost: Cost of any hardware that needs to be purchased, installed, and/or configured in order to implement an architectural recommendation. Usually a one-time fee expressed in dollars.

Position	In-House $/hour	Charge to Client $/hour
IT/Program Manager	50	114
Database Administrator	44	113
System Administrator	32	86
ARCHIBUS Administrator	24	77
Help Desk Person	18	77
Programmer	17	78

Estimates based on 2,080 hours per year

Bi-Monthly – once every two months

Bi-Weekly – once every two weeks; twice a month (semi-monthly)

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
AR-01	All shared drives on the network should enforce authentication policies.	MC-01	High	U	$128	$160	0	0	Implementation Cost: 4.0 Sys Admin $128 Maintenance Cost: 5.0 Sys Admin $160
AR-02	Antivirus software is installed on the server.	MC-17	High	V	$128	$128	$55	0	SW cost: Norton AntiVirus $55 Implementation: 4.0 Sys Admin $128 Maintenance: 4.0 Sys Admin $128
AR-03	Audit information is stored in a separate location from the servers and the workstations.	MC-01, MC-02, MC-03, MC-04, MC-05, MC-06, MC-07	High	U	$320	$160	0	$300	Implementation Cost: 10.0 Sys Admin $320 Maintenance Cost: 5.0 Sys Admin $160 HW cost: Remote/extra disk storage $300

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
AR-04	Block all unnecessary ports at the firewall and host.	MC-17, MC-19	High	P	$128	$128	0	0	Implementation Cost: 4.0 Sys Admin $128 Maintenance Cost: 4.0 Sys Admin $128
AR-05	Check for buffer length.	MC-16	High	P	$356	$172	0	0	Implementation Cost: 7.0 Sys Admin $224 3.0 DBA $132 Maintenance Cost: 4.0 Sys Admin $128 1.0 DBA $44
AR-06	Configure routers to restrict footprinting requests.	MC-19	Medium	W	$128	$128	0	0	Implementation Cost: 4.0 Sys Admin $128 Maintenance Cost: 4.0 Sys Admin $128
AR-07	Database activities should be logged and stored in a separate secure server.	MC-09	Medium	U	$216	$152	0	$300	Implementation Cost: 4.0 Sys Admin $128 2.0 DBA $88 Maintenance Cost: 2.0 Sys Admin $64 2.0 DBA $88

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
									HW cost: Remote/extra disk storage $300
AR-08	Developmental machines should have strong access control mechanisms	MC-08	High	T	$256	$118	0	0	Implementation Cost: 8.0 Sys Admin $256 Maintenance Cost 2.0 Sys Admin $64 3.0 Help Desk $54
AR-09	Disable non-critical services and protocols.	MC-17, MC-19	High	P	$128	$64	0	0	Implementation Cost: 4.0 Sys Admin $128 Maintenance Cost 2.0 Sys Admin $64
AR-10	Display generic information on login screen (e.g., not "loosed-lipped").	MC-20	High	T	$128	$64	0	0	Implementation Cost: 4.0 Sys Admin $128 Maintenance Cost 2.0 Sys Admin $64
AR-11	Dynamic content and force pages to not	MC-12	Medium	T	$230	$98	0	0	Implementation Cost: 4.0 Sys Admin

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
	save cache locally.								$128 6.0 Programmer $102 Maintenance Cost 2.0 Sys Admin $64 2.0 Programmer $34
AR-12	Encrypt user credentials in configurations and databases.	MC-10	High	T	$3040	$760	0	0	Implementation Cost: 40.0 Sys Admin $1280 40.0 DBA $1760 Maintenance Cost 10.0 Sys Admin $320 10.0 DBA $440
AR-13	Ensure that character encoding is set correctly to limit how input can be represented.	MC-22	High	P	$278	$122	0	0	Implementation Cost: 4.0 DBA $176 6.0 Programmer $102 Maintenance Cost

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
									2.0 DBA $88
									2.0 Programmer $34
AR-14	Handle any log exceptions that are allowed to propagate to the application boundary.	MC-22	High	P	$1348	$186	0	0	Implementation Cost: 4.0 Sys Admin $128 20.0 DBA $880 20.0 Programmer $340 Maintenance Cost 2.0 Sys Admin $64 2.0 DBA $88 2.0 Programmer $34
AR-15	Harden weak default configuration setting.	MC-17	High	P	$320	$128	0	0	Implementation Cost: 10.0 Sys Admin $320 Maintenance Cost 4.0 Sys Admin $128
AR-16	Hide HTML source code.	MC-16	High	T	$468	$98	0	0	Implementation Cost: 4.0 Sys Admin $128

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
									20. 0 Programmer $340 Maintenance Cost 2.0 Sys Admin $64 2.0 Programmer $34 Implementation Cost: 4.0 Sys Admin $128 Maintenance Cost 2.0 Sys Admin $64 4.0 Help Desk $72
AR-17	Implement account lock-out policies.	MC-20	High	P	$128	$136	0	0	
AR-18	Implement hierarchical authorization levels.	MC-11	Medium	P	$808	$234	0	0	Implementation Cost: 4.0 Sys Admin $128 8.0 DBA $352 8.0 ARCHIBUS Admin $192

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
									8.0 Programmer $136
									Maintenance Cost
									2.0 Sys Admin $64
									2.0 DBA $88
									2.0 ARCHIBUS Admin $48
									2.0 Programmer $34
		MC-01, MC-02, MC-03, MC-06, MC-07, MC-08, MC-09, MC-10,							Implementation Cost: 4.0 Sys Admin $128 8.0 DBA $352 8.0 ARCHIBUS Admin $192 8.0 Programmer $136
AR-19	Implement role-based authentication.	MC-14	High	U,P	$808	$234	0	0	Maintenance Cost 2.0 Sys Admin

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
									$64 2.0 DBA $88 2.0 ARCHIBUS Admin $48 2.0 Programmer $34
AR-20	Install software-based firewalls on all systems in the network.	MC-19	Medium	P	$512	$400	0	0	Implementation Cost: 16.0 Sys Admin $512 Maintenance Cost: 8.0 Sys Admin $256 8.0 Help Desk $144
AR-21	Invest in backup IT hardware to ensure business continuity.	MC-21	High	D	$688	$256	$1050	$6050	Implementation Cost: 16.0 Sys Admin $512 4.0 DBA $176 Maintenance Cost: 8.0 Sys Admin

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
									$256 HW Cost: Additional Server $5000 Additional Switch (24 10/100 port + 2 1000 port) $1000 Additional Server NIC $ 50 SW Cost: Server license $1000 AntiVirus $ 50 Implementation Cost: 16.0 Sys Admin $512 Maintenance Cost: 8.0 Sys Admin $256
AR-22	Invest in backup network capacity to avoid network downtime and system unavailability.	MC-21	High	D	$512	$256	0	1100	

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
									HW Cost: Additional Server NICs (2) $100 Additional Switch for Server Rm (24 -10/100 port + 2- 1000 port) $1000 Replace all hubs w/ switches with 100Mb ports Upgrade cabling to at least Cat 5, fiber between flrs Upgrade user NICs to 100Mbyte Redundant router paths if subnetted
AR-23	Keep custom configuration stores outside of the Web space.	MC-22	High	P	$320	$64	0	0	Implementation Cost: 10.0 Sys Admin $320 Maintenance Cost: 2.0 Sys Admin $ 64

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
AR-24	Return generic, harmless error messages to the client.	MC-22	High	P	$166	$66	0	0	Implementation Cost: 2.0 Sys Admin $ 64 6.0 Programmer $102 Maintenance Cost: 1.0 Sys Admin $32 2.0 Programmer $34
AR-25	Secure communication channels between servers and servers.	MC-13, MC-19	High	W	$320	64	0	$650	Implementation Cost: 10.0 Sys Admin $320 Maintenance Cost: 2.0 Sys Admin $ 64 HW Cost Switch 12 port 10/100 secure $650
AR-26	Set up firewalls with filtering rules	MC-17, MC-18	High	P	$256	$128	0	$650	Implementation Cost: 8.0 Sys Admin

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
	between servers and workstations.								$256 Maintenance Cost: 4.0 Sys Admin $128 HW Cost 4 port firewall $650
									Implementation Cost: 10.0 Sys Admin $320 Maintenance Cost: 4.0 Sys Admin $128 HW Cost Network Intrusion Detection System $2000
AR-27	Set up an intrusion detection system.	MC-18, MC-21	High	P	$320	$128	0	$2000	
AR-28	Set up IIS to prompt for user credentials every time.	MC-12	Medium	P	$128	$100	0	0	Implementation Cost: 4.0 Sys Admin $128

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
									Maintenance Cost: 2.0 Sys Admin $64 2.0 Help Desk $36
									Implementation Cost: 4.0 Sys Admin $128
AR-29	Shorten the timeout for session kept-alive.	MC-12	Medium	T	$128	$100	0	0	Maintenance Cost: 2.0 Sys Admin $64 2.0 Help Desk $36
									Implementation Cost: 8.0 DBA $352 20.0 Programmer $340
AR-30	Use exception handling through your application's code base.	MC-22	High	P	$692	$346	0	0	Maintenance Cost: 4.0 DBA $176

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
AR-31	Use HTMLEncode and URLEncode functions to encode any HTML output that includes user input.	MC-22	High	P	$686	$129	0	0	10.0 Programmer $170 — Implementation Cost: 4.0 DBA $176; 30.0 Programmer $510 — Maintenance Cost: 1.0 DBA $44; 5.0 Programmer $85
AR-32	Use HTTPS for server-to-client Web data transfer encryption.	MC-12, MC-13, MC-19	High	W	$512	$100	0	0	Implementation Cost: 16.0 Sys Admin $512 — Maintenance Cost: 2.0 Sys Admin $64; 2.0 Help Desk $36

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
									SW Cost: Generate own Windows SSL key and apply to intra-site $ 0
									Implementation Cost: 1.0 Sys Admin |$ 32 3.0 DBA $132 2.0 Programmer $ 34
									Maintenance Cost: 1.0 DBA $ 44 1.0 programmer $ 17
AR-33	Use least privileged account to access the database.	MC-16	High	S	$198	$61	0	0	
									Implementation Cost: 1.0 Sys Admin $ 32 10.0 DBA $440 10.0 Programmer $170
AR-34	Use parameterized stored procedure for database access.	MC-16	High	P	$642	$305	0	0	

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
									Maintenance Cost: 5.0 DBA $220 5.0 programmer $ 85
AR-35	Use regular expressions to make sure file names are well formed.	MC-22	High	P	$100	$17	0	0	Implementation Cost: 1.0 Sys Admin $ 32 4.0 Programmer $ 68 Maintenance Cost: 1.0 Programmer $ 17
AR-36	Use regular expressions to perform thorough input validation.	MC-16, MC-22	High	P	$480	$112	0	0	Implementation Cost: 1.0 Sys Admin $ 32 16.0 Programmer $272 4.0 DBA $176 Maintenance Cost:

No.	Architectural Recommendation	Related Misuse Cases	Priority	Category of Threat	Implementation Cost ($/year)	Maintenance Cost ($/year)	Software Cost [Type]/($)	Hardware Cost [Type]/($)	Acme's Internal Rates
									1.0 DBA $ 44 4.0 Programmer $ 68

Appendix L   Policy Recommendation Costs

Legend:

D: Denial of service

P: System penetration

S: Sabotage of data

T: Theft of proprietary info

U: Unauthorized access by insiders

V: Virus

W: Active Wiretap/Network Eavesdropping

Terms:

Category of Threat: Set of related misuses and attacks that pose threat(s) to the organization.

Training Cost: Cost needed to educate and train users in the organization about how to correctly implement and enforce a policy recommendation. Could also include training material costs (documents, manuals, etc.) and any other follow-up training sessions needed. Usually a one-time fee expressed in dollars.

Enforcement Cost: Cost needed to enforce a policy recommendation after implementation. Includes cost of time spent on enforcing recommendation. It is expressed in dollars per year.

Other Costs: Costs that are specific to the policy recommendation and do not fall under training or enforcement. Could include cost of additional hardware or software. Could be either expressed in dollars or dollars per year, depending on the type.

Position	In-House $/hour	Charge to Client $/hour
IT/Program Manager	50	114
Database Administrator	44	113
System Administrator	32	86
ARCHIBUS Administrator	24	77
Help Desk Person	18	77
Programmer	17	78

Estimates based on 2,080 hours per year

Bi-Monthly – once every two months

Bi-Weekly – once every two weeks; twice a month (semi-monthly)

No.	Policy Recommendation	Related Misuse Cases	Priority	Category of Threat	Training Cost ($)	Enforcement Cost ($)	Other Costs [Type]/ ($)	Total Cost ($/year)	Acme's Internal Rates
PR-01	All installations must be approved and reviewed by managers.	MC-13, MC-15	High	U, W	0	$1968	0	$1968	Enforcement Costs per Month: 2.0 Manager $100 2.0 Sys Admin $ 64
PR-02	Applications and operating systems must be patched routinely. (Bi-Monthly)	MC-01, MC-03, MC-13, MC-15, MC-16, MC-17, MC-18, MC-19, MC-20, MC-21, MC-22	High	U, P	0	$1512	0	$1512	Enforcement Cost bi-monthly: 5.0 Sys Admin $160 2.0 ARCHIBUS Admin $ 48 1.0 DBA $44
PR-03	Audit information must be reviewed routinely. (Monthly)	MC-01, MC-02, MC-03, MC-04, MC-05, MC-06, MC-07, MC-08, MC-09, MC-10, MC-11, MC-12, MC-16, MC-18, MC-22	High	U, P	0	$1536	0	$1536	Enforcement Costs monthly: 4.0 Sys Admin $128

No.	Policy Recommendation	Related Misuse Cases	Priority	Category of Threat	Training Cost ($)	Enforcement Cost ($)	Other Costs [Type]/ ($)	Total Cost ($/year)	Acme's Internal Rates
PR-04	Configuration changes are stored and cross-reviewed. (Monthly)	MC-02, MC-03, MC-04, MC-05, MC-06, MC-09, MC-16, MC-17, MC-22	High	U, P	0	$1584	0	$1584	Enforcement Costs monthly: 2.0 Sys Admin $64 1.0 DBA $44 1.0 ARCHIBUS Admin $24
PR-05	Develop disaster recovery contingency plan.	MC-21, MC-17	High	D,V	$3000	$21720	$6050	$30770	Enforcement Costs per month: 20.0 Sys Admin $640 10.0 DBA $440 5.0 ARCHIBUS Admin $120 5.0 Manager $250 2.0 Help Desk $360 Training costs per year: Documentation $3000 Other Costs per year: Redundant/spare HW $6050
PR-06	Do not set up shared files/folders/drives on the AMS network server or workstation.	MC-08, MC-10, MC-11, MC-14	High	U, W	0	$384	0	$384	Enforcement Costs monthly : 1.0 Sys Admin $32
PR-07	Enforce strong	MC-01, MC-	High	U	$300	$408	0	$708	Enforcement Costs per month:

No.	Policy Recommendation	Related Misuse Cases	Priority	Category of Threat	Training Cost ($)	Enforcement Cost ($)	Other Costs [Type]/ ($)	Total Cost ($/year)	Acme's Internal Rates
	password policies.	03, MC-08, MC-10, MC-11, MC-20							0.5 Sys Admin $16 1.0 Help Desk $18 Training Costs per year: $300
PR-08	Firewalls and IDS must be patched routinely. (Monthly)	MC-13, MC-17, MC-18, MC-19, MC-21	High	P	0	$1536	0	$1536	Enforcement Costs per month: 4.0 Sys Admin $128
PR-09	Follow the principle of least privilege and use least privileged service accounts to run processes and access resources.	MC-08, MC-09, MC-11, MC-17, MC-20, MC-22	High	U	0	$1296	0	$1296	Enforcement Costs per month: 2.0 Sys Admin $64 1.0 DBA $44
PR-10	Log all incoming and outgoing traffic (IIS, database engine, MapGuide, firewall).	MC-08, MC-13, MC-17, MC-18, MC-19	High	P	0	$1680	$300	$1980	Enforcement Costs per month: 3.0 Sys Admin $96 1.0 DBA $44 Other costs (one time): Remote/extra disk storage $300

No.	Policy Recommendation	Related Misuse Cases	Priority	Category of Threat	Training Cost ($)	Enforcement Cost ($)	Other Costs [Type]/ ($)	Total Cost ($/year)	Acme's Internal Rates
PR-11	New systems on the network should be evaluated prior to deployment.	MC-17	High	V	0	$1092	0	$1092	Enforcement Costs per month: 2.0 Sys Admin $64 1.5 Help Desk $27
PR-12	Only sys admins are permitted to install any software and/or hardware.	MC-13, MC-15	High	T, U	0	$984	0	$984	Enforcement Costs per month: 2.0 Sys Admin $64 1.0 Help Desk $18
PR-13	Password-protect any necessary shared documents.	MC-01, MC-03, MC-06, MC-08, MC-09, MC-10, MC-11, MC-13, MC-14	High	U	$300	$1368	0	$1668	Enforcement Costs per month: 3.0 Sys Admin $96 1.0 Help Desk $18 Training Costs per year: $300
PR-14	Perform information integrity checks on a routine basis, reviewed by DBA and/or managers. (Bi-Weekly)	MC-06, MC-07, MC-08, MC-09, MC-10, MC-14, MC-15	High	T, U	0	$5096	0	$5096	Enforcement costs per bi-weekly: 2.0 Sys Admin $64 3.0 DBA $132
PR-15	Perform routine code review. (Monthly)	MC-16	High	U	0	$1728	0	$1728	Enforcement costs per month: 1.0 Sys Admin $32

No.	Policy Recommendation	Related Misuse Cases	Priority	Category of Threat	Training Cost ($)	Enforcement Cost ($)	Other Costs [Type]/ ($)	Total Cost ($/year)	Acme's Internal Rates
							0		4.0 Programmer $68 1.0 DBA $44
PR-16	Require users to change their passwords periodically. (Monthly)	MC-01, MC-03, MC-10, MC-20	High	U	$300	$408	0	$708	Enforcement Costs per month: 0.5 Sys Admin $16 1.0 Help Desk $18 Training Costs per year: $300
PR-17	Routers must be patched routinely. (Monthly)	MC-19	Medium	W	0	$192	0	$192	Enforcement costs per month: 0.5 Sys Admin $16
PR-18	Separate personnel review of sys admin's activities. (Monthly)	MC-02, MC-04, MC-05	High	U	0	$3072	0	$3072	Enforcement costs per month: 8.0 Sys Admin $256
PR-19	Set clear and defined user access controls for all users. (Low, Medium, High, System Admins)	MC-01, MC-08, MC-10, MC-11, MC-20	High	U	$300	$1536	0	$1836	Enforcement costs per month: 4.0 Sys Admin $128 Training costs per year: $300
PR-20	Perform routine system and data back-up. (Weekly)	MC-01, MC-02, MC-03, MC-04, MC-	High	V,S	0	$8320	0	$18795	Enforcement costs per week: 5.0 Sys Admin $160

No.	Policy Recommendation	Related Misuse Cases	Priority	Category of Threat	Training Cost ($)	Enforcement Cost ($)	Other Costs [Type]/ ($)	Total Cost ($/year)	Acme's Internal Rates
		05, MC-06, MC-07, MC-08, MC-09, MC-10, MC-14, MC-15, MC-16, MC-17, MC-18, MC-21, MC-22					$10475		Other Costs per year: Backup software Veritas for Windows 2003 server (CDW) $500 Veritas SQL Server agent (CDW) $600 Quantum 300/600GB Backup tapes (CDW) 4 weekly 4 incremental 12 monthly 1 yearly 4 extra (25 tapes @ $135 each) $3375 Fire Safe (Staples) $1000 Offsite Tape Storage Service or Electronic vaulting service

No.	Policy Recommendation	Related Misuse Cases	Priority	Category of Threat	Training Cost ($)	Enforcement Cost ($)	Other Costs [Type]/($)	Total Cost ($/year)	Acme's Internal Rates
									$5000 (?)
									Enforcement costs bi-monthly: 4.0 Sys Admin $128 3.0 DBA $132 3.0 ARCHIBUS Admin $72
PR-21	User activities must be periodically reviewed. (Bi-Monthly)	MC-01, MC-02, MC-03, MC-04, MC-05, MC-06, MC-07, MC-08, MC-09, MC-10, MC-11, MC-12, MC-13, MC-15, MC-18, MC-20, MC-22	High	U	0	$1992	0	$1992	
PR-22	Users should have an automated log out of the AMS system after a certain time of idle activity.	MC-03, MC-06, MC-07, MC-12	High	U	$300	$600		$900	Enforcement Costs per month: 1.0 Sys Admin $32 1.0 Help Desk $18 Training Costs:

No.	Policy Recommendation	Related Misuse Cases	Priority	Category of Threat	Training Cost ($)	Enforcement Cost ($)	Other Costs [Type]/ ($)	Total Cost ($/year)	Acme's Internal Rates
							0		$300
PR-23	Users should not have rights or access levels beyond those prescribed by their job responsibilities.	MC-01, MC-02, MC-03, MC-06, MC-08, MC-09, MC-10, MC-11	High	U	$300	$1800	0	$2100	Enforcement Costs per month: 2.0 Sys Admin $64 2.0 Help Desk $36 1.0 Manager $50
PR-24	Users should not reveal their account names and passwords in any situation.	MC-01, MC-03, MC-10, MC-20	High	U	$300	$192	0	$492	Enforcement Costs per month: 0.5 Sys Admin $16 Training Costs per year: $300

Appendix M   Misuse Case Losses

Legend:

D: Denial of service
P: System penetration
S: Sabotage of data
T: Theft of proprietary info
U: Unauthorized access by insiders
V: Virus
W: Active Wiretap/Network Eavesdropping

Terms:
Fixing Cost: Cost needed to fix the result of a misuse case being exploited. Could include costs of external teams that are hired to solve the problem. Expressed in dollars.

Productivity Loss: Cost of lost productivity when the system or part of the system is non-functional or jeopardized as a result of the misuse case exploitations. Expressed in dollars.

Other Losses: Cost of other losses that do not fall under any of the other categories and are particular to the specific misuse case. Expressed in dollars.

Position	In-House $/hour	Charge to Client $/hour
IT/Program Manager	50	114
Database Administrator	44	113
System Administrator	32	86
ARCHIBUS Administrator	24	77
Help Desk Person	18	77
Programmer	17	78
User	15	

Estimates based on 2,080 hours per year
Bi-Monthly – once every two months
Bi-Weekly – once every two weeks; twice a month (semi-monthly)

No.	Misuse Case Name	Category of Threats	Priority	Fixing Cost ($)	Productivity Loss ($)	Other Losses [Type]/($)	Total Loss ($/year)	Per Incident Loss ($)
MC-01	Unauthorized logon on the Win 2003 server.	U	High	$128	0	0	$128	Fixing Cost: 4.0 Sys Admin $128
MC-03	Users gain Sys Admin rights on the Windows 2003 server. (Elevation of Privilege)	U	High	$128	0	0	$128	Fixing Cost: 4.0 Sys Admin $128
MC-04	Sys Admin deletes critical system configurations on the Windows 2003 server.	S	High	$256	$1184	$2500	$3940	Fixing Cost: 8.0 Sys Admin $256 Productivity Loss: 8.0 Sys Admin $256 8.0 ARCHIBUS Admin $192 8.0 Programmer $136 8.0 Users (5) $600 Other Losses: Loss of Reputation $2500
MC-06	User deletes critical data from	S	High	$264	$412	$2500	$3176	Fixing Cost:

No.	Misuse Case Name	Category of Threats	Priority	Fixing Cost ($)	Productivity Loss ($)	Other Losses [Type]/($)	Total Loss ($/year)	Per Incident Loss ($)
	the AMS system.							2.0 Sys Admin $128 2.0 DBA $ 88 2.0 ARCHIBUS Admin $ 48 Productivity Loss: 2.0 Sys Admin $126 2.0 DBA $ 88 2.0 ARCHIBUS Admin $ 48 2.0 Users (5) $150 Other Losses: Loss of reputation $2500
MC-08	System data is accessed through developmental machines.	U	High	$112	$112	0	$224	Fixing Cost: 2.0 Sys Admin $ 64 2.0 ARCHIBUS Admin $ 48

No.	Misuse Case Name	Category of Threats	Priority	Fixing Cost ($)	Productivity Loss ($)	Other Losses [Type]/($)	Total Loss ($/year)	Per Incident Loss ($)
MC-10	Steal user credential information through developmental machines.	T	High	$112	$112	$2500	$2724	Productivity Cost: 2.0 Sys Admin $64 2.0 ARCHIBUS Admin $48 Fixing Cost: 2.0 Sys Admin $64 2.0 ARCHIBUS Admin $48 Productivity Loss Cost: 2.0 Sys Admin $64 2.0 ARCHIBUS Admin $48 Other Losses: Loss of integrity & confidentiality
MC-13	Malicious users tap communications channel between workstations and servers.	W	High	$256	$256	$2500	$3012	Fixing Cost: 8.0 Sys Admin $256

No.	Misuse Case Name	Category of Threats	Priority	Fixing Cost ($)	Productivity Loss ($)	Other Losses [Type]/($)	Total Loss ($/year)	Per Incident Loss ($)
MC-16	Input Validation Attack	P	High	$256	$256	$5000	$5512	Productivity Loss Cost: 8.0 Sys Admin $256 Other Losses: Loss of reputation $2500 Loss of confidentiality $2500
MC-17	Infect Windows 2003 server	V	High	$256	$1184	$2500	$3940	Fixing Cost: 8.0 Sys Admin $256 Productivity Loss Cost: 8.0 Sys Admin $256 Other Losses: Loss of reputation $2500 Loss of confidentiality $2500 Fixing Cost: 8.0 Sys Admin $256

No.	Misuse Case Name	Category of Threats	Priority	Fixing Cost ($)	Productivity Loss ($)	Other Losses [Type]/($)	Total Loss ($/year)	Per Incident Loss ($)
	with virus/worms.							Productivity Loss: 8.0 Sys Admin $256 8.0 ARCHIBUS Admin $192 8.0 Programmer $136 8.0 Users (5) $600 Other Losses: Loss of reputation $2500
								Fixing Cost: 4.0 Sys Admin $128 Productivity Loss: 4.0 Sys Admin $128 Other Losses: Loss of reputation $2500
MC-20	Brute Force Attacks: Password Cracking/Credential Theft	P	High	$128	$128	$5000	$5256	Loss of integrity &

No.	Misuse Case Name	Category of Threats	Priority	Fixing Cost ($)	Productivity Loss ($)	Other Losses [Type]/($)	Total Loss ($/year)	Per Incident Loss ($)
								confidentiality $2500
								Fixing Cost: 8.0 Sys Admin $256
								Productivity Loss: 8.0 Sys Admin $256 8.0 ARCHIBUS Admin $192 8.0 Programmer $136 8.0 Users (5) $600
								Other Losses: Loss of reputation $2500
MC-21	Denial of Service	D	High	$256	$1184	$2500	$3940	
MC-22	Execute malicious code.	P	High	$256	$1184	$5000	$6440	Fixing Cost: 8.0 Sys Admin $256 Productivity Loss: 8.0 Sys Admin

No.	Misuse Case Name	Category of Threats	Priority	Fixing Cost ($)	Productivity Loss ($)	Other Losses [Type]/($)	Total Loss ($/year)	Per Incident Loss ($)
								$256 8.0 ARCHIBUS Admin $192 8.0 Programmer $136 8.0 Users (5) $600 Other Losses: Loss of reputation $2500 Loss of integrity & confidentiality $2500

References

URLs are valid as of the publication date of this document.

[Alberts 03]	Alberts, Christopher & Dorofee, Audrey. *OCTAVE Threat Profiles*. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2003. http://www.cert.org/archive/pdf/OCTAVEthreatProfiles.pdf.
[Allen 99]	Allen, J.; Christie, A.; Fithen, W.; McHugh, J.; Pickel, J.; & Stoner, E. *State of the Practice of Intrusion Detection Technologies* (CMU/SEI-99-TR-028, ADA375846). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1999. http://www.sei.cmu.edu/publications/documents/99.reports/99tr028 /99tr028abstract.html.
[Ambler 04]	Ambler, Scott W. *Active Stakeholder Participation*. Evergreen, Colorado, Ronin International, Inc., 2004. http://www.agilemodeling.com/essays/activeStakeholderParti cipation.htm.
[ARCHIBUS 02]	ARCHIBUS, Inc. ARCHIBUS/FM®: Solution Integration with Cyco Software AutoManager® Meridian. http://www.cyco.com/download/whitepaper/Cyco_WhitePaper_Arc hibus.pdf (2002).
[ARCHIBUS 04]	ARCHIBUS, Inc. *FM Web Central 14® Enhanced Communication and Collaboration*. http://www.archibus.com/products/wc14features.cfm (2004).
[ASI 04]	Application Security Inc. *Header Manipulation*. New York, N.Y., Application Security Inc., 2004. http://www.appsecinc.com/Policy/PolicyCheck6002.html.
[Brown 05]	Brown, Keith. "Item 7 : What is a Luring Attack?" in *The .NET Developer's Guide to Windows Security*. Boston, MA: Addison-Wesley, 2005. http://www.pluralsight.com/keith/book/html/whatis_luring.html.

[CERT/CC 01]	CERT Coordination Center. *Monitor and inspect system activities for unexpected behavior*. http://www.cert.org/security-improvement/practices/p095.html (2001).
[CERT/CC 04]	CERT Coordination Center. *Responding to Intrusions*. http://www.cert.org/security-improvement/modules/m06.html (2004).
[Computer Hope 04]	Computer Hope. *DOS*. West Jordan, UT, Computer Hope, 2004. http://www.computerhope.com/jargon/d/dos.htm.
[Dictionary.com 04a]	Dictionary.com. *Disclosure*. Los Angeles, CA., Lexico Publishing Group, LLC, 2004. http://dictionary.reference.com/search?q=disclosure.
[Dictionary.com 04b]	Dictionary.com. *Espionage*. Los Angeles, CA., Lexico Publishing Group, LLC, 2004. http://dictionary.reference.com/search?q=espionage.
[Dictionary.com 04c]	Dictionary.com. *Privacy*. Los Angeles, CA., Lexico Publishing Group, LLC, 2004. http://dictionary.reference.com/search?q=privacy.
[Ellison 97]	Ellison, B.; Fisher, D. A.; Linger, R. C.; Lipson, H. F.; Longstaff, T.; & Mead, N. R. *Survivable Network Systems: An Emerging Discipline* (CMU/SEI-97-TR-013, ADA341963). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 1997. http://www.sei.cmu.edu/publications/documents/97.reports/97tr013/97tr013abstract.html.
[Ellison 03]	Ellison, R. & Moore, A. *Trustworthy Refinement Through Intrusion-Aware Design* (CMU/SEI-2003-TR-002, ADA414865). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2003. http://www.sei.cmu.edu/publications/documents/03.reports/03tr002.html.
[Farlex 04]	Farlex, Inc. "Man in the middle attack." *TheFreeDictionary.com*. http://encyclopedia.thefreedictionary.com/man%20in%20the%20middle%20attack (2004).

[FFIEC 04]	FFIEC. "Booklet: Information Security Section: Appendix B: Glossary." Washington, D.C., Federal Financial Institutions Examination Council, 2004. http://www.ffiec.gov/ffiecinfobase/booklets/information_security/08_glossary.html.
[Franklin 04]	Franklin, Curtis, Jr. *Setting Up an Intrusion Detection System*. http://www.securitypipeline.com/network/showArticle.jhtml?articleId=22104387&pgno=1 (2004).
[Guttman 95]	Guttman, Barbara; Roback, Edward. *An Introduction to Computer Security*. Gaithersburg, MD: U.S. Department of Commerce, Technology Administration, National Institute of Standards and Technology, 1995. http://csrc.nist.gov/publications/nistpubs/800-12/handbook.pdf.
[Howard 97]	Howard, John D. *An Analysis of Security Incidents on the Internet 1989-1995*. Pittsburgh, Pa., Carnegie Mellon University, Software Engineering Institute, 1997. http://www.cert.org/research/JHThesis/Start.html.
[Howard 98]	Howard, John; Longstaff, Thomas. *A Common Language for Computer Security Incidents*. Albuquerque, N.M., Sandia National Laboratories, 1998. http://www.cert.org/research/taxonomy_988667.pdf.
[Humpherys 04]	Humpherys, Jeffrey. *Using Server.URLEncode*. http://www.4guysfromrolla.com/webtech/042601-1.shtml (2004).
[Mead 03]	Mead, Nancy. *Requirements Engineering for Survivable Systems* (CMU/SEI-2003-TN-013, ADA418410). Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2003. http://www.sei.cmu.edu/publications/documents/03.reports/03tn013.html
[Meier 03]	Meier, J. D.; Mackman, Alex; Vasireddy, Srinath; Dunner, Michael; Escamilla, Ray; & Murukan, Anandha. *Improving Web Application Security: Threats and Countermeasures* Chapter 2, "Threats and Countermeasures." Redmond, Washington, Microsoft, 2003 http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnnetsec/html/ThreatCounter.asp.

[Microsoft 03a]	Microsoft Corporation. *Auditing Security Events—Best Practices*. http://www.microsoft.com/resources/documentation/WindowsServ /2003/standard/proddocs/en-us/Default.asp?url=/resources /documentation/windowsserv/2003/standard/proddocs /en-us/sag_seconceptsimpaudbp.asp (2003).
[Microsoft 03b]	Microsoft Corporation. *Internet Information Services 6.0 Features*. http://www.microsoft.com/windowsserver2003/iis /evaluation/features/default.mspx (2003).
[Microsoft 03c]	Microsoft Corporation. *Best practices for permissions and user rights*. http://www.microsoft.com/resources/documentation/WindowsServ/ 2003/standard/proddocs/en-us/Default.asp?url= /resources/documentation/windowsserv/2003/standard/proddocs /en-us/sag_SEconceptsImpACBP.asp (2003).
[Microsoft 03d]	Microsoft Corporation. *Role-Based Access Control Using Windows Server 2003 Authorization Manager*. http://msdn.microsoft.com/library/default.asp?url=/library /en-us/dnnetserv/html/AzManRoles.asp (2003).
[Microsoft 03e]	Microsoft Corporation. *Using Microsoft Windows IPSec to Help Secure an Internal Corporate Network Server*. http://www.microsoft.com/downloads/details.aspx?FamilyID=a774 012a-ac25-4a1d-8851-b7a09e3f1dc9&DisplayLang=en (2003).
[Microsoft 04a]	Microsoft Corporation. *Securing Your Network*. http://msdn.microsoft.com/library/default.asp?url=/library /en-us/secmod/html/secmod88.asp (2004).
[Microsoft 04b]	Microsoft Corporation. *How To Implement Forms-Based Authentication in Your ASP.NET Application by Using C# .NET*. http://support.microsoft.com/default.aspx?scid=kb;en-us;Q301240&ID=kb;en-us;Q301240&SD=MSDN (2004).
[Microsoft 04c]	Microsoft Corporation. *ASP Built-in Objects*. http://msdn.microsoft.com/library/default.asp?url=/library/en-us/iissdk/iis/ref_vbom_.asp (2004).
[Microsoft 04d]	Microsoft Corporation. *Securing Your Web Server*. http://msdn.microsoft.com/library/default.asp?url=/library/en-us/secmod/html/secmod89.asp (2004).

[Microsoft 04e]	Microsoft Corporation. *IIS 6.0 Resource Guide Tools*. http://www.microsoft.com/resources/documentation/IIS/6/all/techref/en-us/iisRG_IIS_7.mspx (2004).
[Microsoft 04f]	Microsoft Corporation. *How To Set Up an HTTPS Service in IIS*. http://support.microsoft.com/default.aspx?scid=kb;en-us;324069 (2004).
[Microsoft 04g]	Microsoft Corporation. *.NET Framework Regular Expressions*. http://msdn.microsoft.com/library/default.asp?url=/library/en-us/cpguide/html/cpconCOMRegularExpressions.asp (2004).
[NIPC 02]	NIPC Special Technologies and Applications Unit (STAU). *Insiders and Information Technology*. Washington, D.C.: Department of Homeland Security, Information Analysis Infrastructure Protection, 2002. http://www.hpcc-usa.org/pics/02-pres/wright.ppt.
[Pfleeger 03]	Pfleeger, Charles P. & Pfleeger, Shari Lawrence. *Security in Computing*. 3rd edition. Upper Saddle River, NJ: Prentice Hall PTR, 2003.
[RUsecure 04a]	RUsecure . "Controls." *Information Security Glossary*. http://www.yourwindow.to/information-security/gl_controls.htm (2004).
[RUsecure 04b]	RUsecure. "Downtime." *Information Security Glossary*. http://www.yourwindow.to/information-security/gl_downtime.htm (2004).
[RUsecure 04c]	RUsecure. "Penetration." *Information Security Glossary*. http://www.yourwindow.to/information-security/gl_penetration.htm (2004).
[RUsecure 04d]	RUsecure. "Penetration Testing." *Information Security Glossary*. http://www.yourwindow.to/information-security/gl_penetration.htm (2004).
[RUsecure 04e]	RUsecure. "Resilience." *Information Security Glossary*. http://www.yourwindow.to/information-security/gl_resilience.htm (2004).
[RUsecure 04f]	RUsecure. "Toolkit." *Information Security Glossary*. http://www.yourwindow.to/information-security/gl_toolkit.htm (2004).

[SANS 03a]	The SANS Institute. *SANS Glossary of Terms Used in Security and Intrusion Detection*. http://www.sans.org/resources/glossary.php#top (2003).
[SANS 03b]	The SANS Institute. *SANS Glossary of Terms Used in Security and Intrusion Detection*. http://www.sans.org/resources/glossary.php#S (2003).
[SANS 04]	The SANS Institute. *Password Policy*. http://www.sans.org/resources/policies/Password_Policy.pdf (accessed December 16, 2004).
[Schorr 04]	Schorr, Joseph Patrick. *Configuring Intrusion Detection in ISA Server*. http://www.isaserver.org/tutorials/Configuring_Intrusion_Detection_in_ISA_Server.html (2004).
[scriptasylum 04]	scriptasylum.com. *HTML/text/JavaSript Escaping/Encoding Script*. http://scriptasylum.com/tutorials/encdec/encode-decode.html (accessed on December 14, 2004).
[Shinder 04a]	Shinder, Thomas. *ISA Server Security Checklist—Part 2: Securing the ISA Server Configuration*. http://www.isaserver.org/tutorials/ISA_Server_Security_Checklist__Part_2_Securing_the_ISA_Server_Configuration.html (2004).
[Shinder 04b]	Shinder, Thomas. *Configuring Fault Tolerance and Load Balancing for Windows 2003 ISA Firewall/VPN Servers*. http://www.isaserver.org/pages/article.asp?id=1114 (2004).
[Shostak 04]	Shostak, Adam. *Security Code Review Guidelines*. http://www.homeport.org/~adam/review.html (2004).
[SSI 03]	Service Strategies Inc. "Nonrepudiation." *Glossary of Messaging & Security Terms*. http://www.ssimail.com/Glossary.htm#N (2003).
[Sybase 97]	Sybase, Inc. *Sybase Central for Adaptive Server Enterprise*. http://www.sybase.com/detail?id=20174 (1997).
[Sybase 01]	Sybase, Inc. *Sybase® Enterprise Portal™ Security*. http://www.sybase.com/detail?id=1012059 (2001).

[Sybase 03]	Sybase, Inc. *New Security Features in Sybase® Adaptive Server® Enterprise*. http://www.sybase.com/content/1013009/new_security_wp.pdf (2003).
[TechTarget 03a]	TechTarget. *Access Control List*. http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci213757,00.html (2003).
[TechTarget 03b]	TechTarget. *Authentication*. http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci211621,00.html (2003).
[TechTarget 03c]	TechTarget. *Encryption*. http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci212062,00.html (2003).
[TechTarget 03d]	TechTarget. *Fault-Tolerant*. http://whatis.techtarget.com/definition/0,,sid9_gci214456,00.html (2003).
[TechTarget 03e]	TechTarget. *Malware*. http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci762187,00.html (2003).
[TechTarget 03f]	TechTarget. *Script Kiddy*. http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci550928,00.html (2003).
[TechTarget 04a]	TechTarget. *Anti-Virus Software*. http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci211573,00.html (2004).
[TechTarget 04b]	TechTarget. *Cracker*. http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci211852,00.html (2004).
[TechTarget 04c]	TechTarget. *Disaster Recovery Plan*. http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci752089,00.html (2004).
[TechTarget 04d]	TechTarget. *Distributed Denial-of-Service Attack*. http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci557336,00.html (2004).

[TechTarget 04e]   TechTarget. *Trojan Horse*.
http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci213221
,00.html (2004).

[TechTarget 04f]   TechTarget. *Virus*.
http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci213306
,00.html (2004).

[TechTarget 04g]   TechTarget. *Worm*.
http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci213386
,00.html (2004).

[Tittel 03]   Tittel, Ed. *Security audit action list for CIOs*.
http://techrepublic.com.com/5100-6296-5054775.html (2003).

[Webopedia 04a]   Webopedia.com. *Firewall*.
http://www.webopedia.com/TERM/F/firewall.html (2004).

[Webopedia 04b]   Webopedia.com. *Malware*.
http://www.webopedia.com/TERM/M/malware.html (2004).

[Webopedia 04c]   Webopedia.com. *Port Scanning*.
http://www.webopedia.com/TERM/P/port_scanning.html (2004).

[Webopedia 04d]   Webopedia.com. *Spoof*.
http://www.webopedia.com/TERM/S/spoof.html (2004).

[West-Brown 03]   West-Brown, M.; Stikvoort, D.; Kossakowski, K.; Killcrece, G.;
Ruefle, R.; & Zajicek, M. *Handbook for Computer Security
Incident Response Teams (CSIRTs)* (CMU/SEI-2003-HB-002).
Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon
University, 2003.
http://www.sei.cmu.edu/publications/documents/03.reports/03hb00
2.html.

[Wikipedia 04]   Wikipedia. *Intrusion-Prevention System*.
http://en.wikipedia.org/wiki/Intrusion-prevention_system (2004).

[Zamir 04]   Zamir, Liran. *Allowing Norton AntiVirus software LiveUpdate
through ISA Server*.
http://www.isaserver.org/tutorials/Allowing_Norton_AntiVirus_soft
ware_LiveUpdate_through_ISA_Server.html (2004).

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
(Leave Blank)	December 2004	Final

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
System Quality Requirements Engineering (SQUARE) Methodology: Case Study on Asset Management System	F19628-00-C-0003

6. AUTHOR(S)

Peter Chen, Marjon Dean, Don Ojoko-Adams, Hassan Osman, Lilian Lopez, Nick Xie

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Software Engineering Institute Carnegie Mellon University Pittsburgh, PA 15213	CMU/SEI-2004-SR-015

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
HQ ESC/XPK 5 Eglin Street Hanscom AFB, MA 01731-2116	

11. SUPPLEMENTARY NOTES

12A DISTRIBUTION/AVAILABILITY STATEMENT	12B DISTRIBUTION CODE
Unclassified/Unlimited, DTIC, NTIS	

13. ABSTRACT (MAXIMUM 200 WORDS)

This report exemplifies the application of the System Quality Requirements Engineering (SQUARE) methodology developed by the Software Engineering Institute's Networked Systems Survivability Program on an asset management application. An overview of the SQUARE process and the vendor is presented, followed by a description of the application under study. The nine-step process of requirements engineering is then explained, and feedback on its implementation is provided. The report concludes with a synopsis of the findings and recommendations for future work.

This report is one of a series of reports resulting from research conducted by the SQUARE Team as part of an independent research and development project of the Software Engineering Institute.

14. SUBJECT TERMS	15. NUMBER OF PAGES
information security improvement, information security costs, misuse cases, requirements engineering, system survivability	327

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89) Prescribed by ANSI Std. Z39-18 298-102