GIAC
CERTIFICATIONS

Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permited without express written permission.

Implementing an Information Security Management System in an Internal Web Development Environment

GIAC ISO-17799 Certification (G7799)
Practical Assignment – Version 1.1
SANS 2004 (Orlando, FL)

Joseph McComb
October 28th, 2004

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Table of Contents

Abstract 3
I. The System Defined 3
The Company 3
The Origin of the Environment 6
The Current Environment 7
Current Web Applications and Sites in the Environment 10
Current State of Security 12
Scope of Information Security Management System (ISMS) 15
II. Planning the Implementation of the Information Security Management System (ISMS) 15
Management Structure 15
The Asset Inventory 18
Policies 21
Risk Identification and Analysis Process 23
Plans for Risk Management 24
III. Implementation (the "Do" phase) 33
Correcting the Problems Identified in the Risk Management Plan 33
Statements Of Applicability 43
IV. Check – System Auditing 44
V. Continuous Improvement ("Act" Phase) 51
Improving the System Through Lessons Learned from Incident Handling 51
Improving the System through Auditing 51
Bibliography 52
Appendix A – Extended Asset Classification 53
Appendix B – Policies 62
Policy – System and Application Access Control (section 9.1 of the ISO 17799 standard) 62
Policy – Business Continuity Planning (section 11.1 of the ISO 17799 standard) 63
Policy – Security Engineering in the Systems Development Life Cycle (section 10.1 of the ISO 17799 standard) 64
Appendix C – Fault Tree Analysis 65
Appendix D – Flagged System Events 657
Appendix E – High Level Plan for Risk Management 81
Appendix F – Extended Audit Checklist 82

Table of Figures

Figure 1. Overview of the Drug Development Stages 5
Figure 2. Diagram of the Web Server Environment 8
Figure 3. Overview of the Systems Development Life Cycle 10
Figure 4. Information Flow in the Data Center Environment 11
Figure 5. Information Flow in the Development Environment 12

Table of Tables

Table 1. Plan for Risk Management 26
Table 2. Documentation of System Problems. 33
Table 3. Audit Checklist for User Access Management 45

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Abstract

A pharmaceutical company has created an internal web development team that creates web sites and applications to serve the departments of the research division. The web development unit maintains a development and production environment for their web sites. The entire environment is behind the company firewall and serves only internal users. Because proprietary information is stored within the environment, the upper management has requested that ISO 17799 be implemented to improve system security. This paper details the implementation of ISO 17799 in the web development environment including the plan for risk management, the controls selected and audit checklist created to audit the effectiveness of the controls.

I. The System Defined

The Company

The research division of a pharmaceutical company has recently decided to pilot an ISO 17799 program to improve security of the production web server environment that is managed by the divisional web development unit. The upper management has become concerned about the storage of proprietary information on the web server environment and has requested that the web server environment move towards ISO 17799 compliance. The web development team believes that this is a reasonable move because a recent security audit has revealed weaknesses in the environment.

Organizationally, the company is broken into different functional divisions:

1. The research division discovers new compounds and moves the compounds through the clinical trials until they receive FDA approval.
2. The manufacturing division produces the drugs for market.
3. The marketing division handles the marketing and sales of the drugs.
4. The corporate infrastructure division handles the shared operations of the company, such as payroll and the network infrastructure. The corporate infrastructure division manages the company firewall.

Currently, the company has approximately 35,000 employees spread across the world, with 5,000 employees in the research division. The company has a global presence with research and marketing sites across the world. Most of the sites are small and possess less than 100 employees. The company also possesses several large sites that house several thousand employees apiece. The larger sites are contained in Europe and North America.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

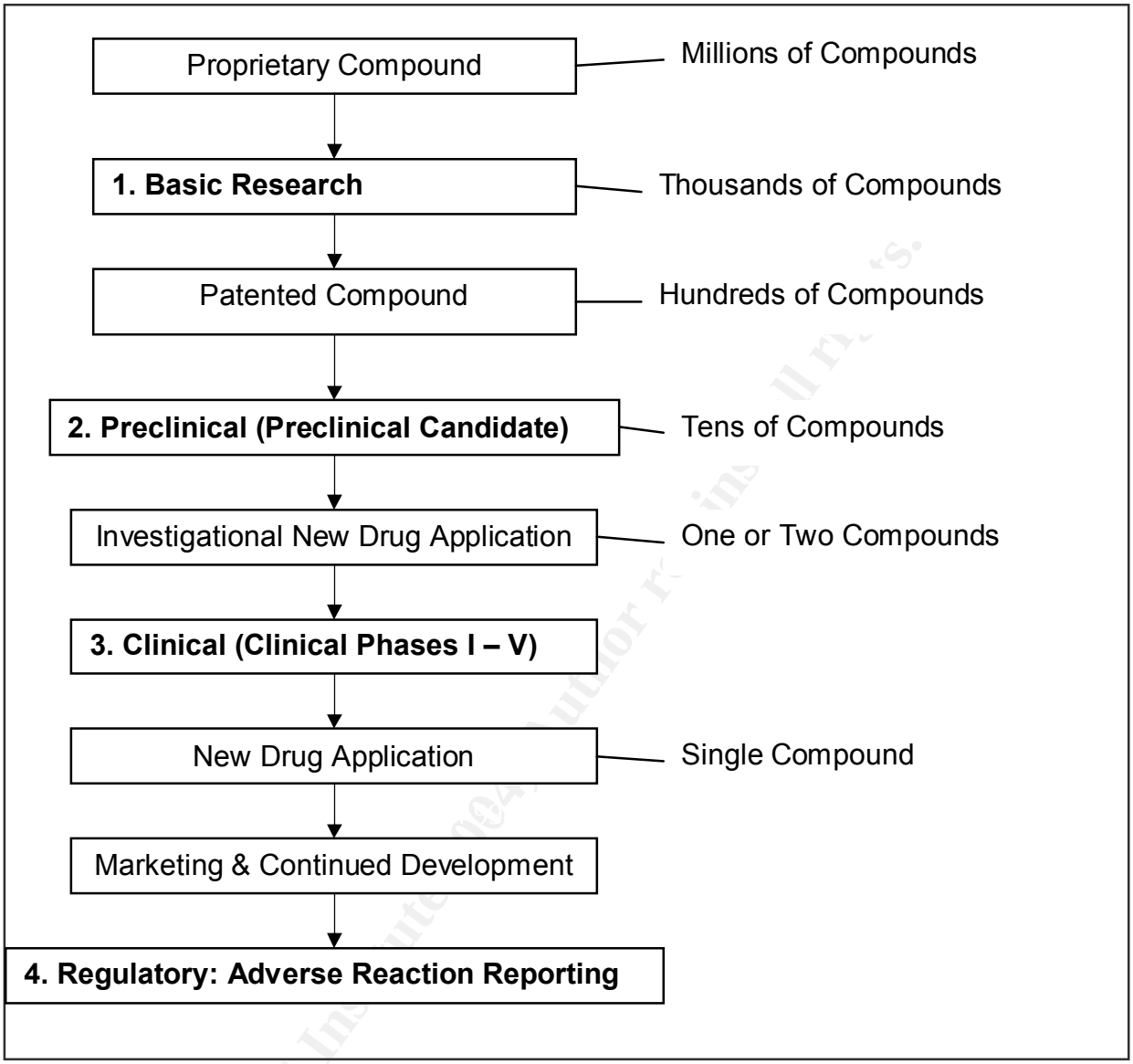
The core business of the company is to research new medicinal compounds and to bring them to market. The drug development cycle[1] follows the diagram shown in Figure 1. Essentially, the business maintains proprietary information on millions of compounds. In the basic research phase, compounds are selected that show promise to have an effect upon a specific medical condition. Through testing, compounds are excluded until certain promising families of compounds are left. Typically, these compounds are patented to protect the rights of the company. The basic research phase continues until only a few compounds remain. These remaining compounds, called Pre-Clinical Candidates, are tested for toxicity and efficacy. Over the next couple of years, compounds that prove to be ineffective or too toxic are eliminated. Soon only one or two related compounds remain for that particular medical condition. The company then files an investigational new drug application to move the drug into clinical trials. Over the next several years, the compounds are tested in human trials. If the compounds are proven effective and have few side effects, the company will file a new drug application with the FDA. If approved, the drug will proceed to marketing and the company will continue to receive adverse reactions of patients taking the drug to monitor the safety of the compound. All total, the process encompasses ten to fifteen years of research and typically costs hundreds of millions of dollars[2].

[1] A good freely available reference (Pharmaceutical Industry Profile 2004) on the drug development process can be found at the Pharmaceutical Manufactures Association at: http://www.phrma.org/publications/publications//2004-03-31.937.pdf

[2] On average the cost to develop a drug is $800 million dollars. See Why Do Prescription Drugs Cost So Much? and Other Questions About Your Medicines (2002-2004) at http://www.phrma.org/publications/publications/brochure/questions/whycostmuch.cfm

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Figure 1. Overview of the Drug Development Stages



The research division relies heavily on information technology. Most employees have a personal desktop unit with access to e-mail and the Microsoft Office suite. The research division uses a standard Windows-based desktop and utilizes several hundred custom and commercial applications. Most laboratory equipment is attached to a PC so that data can be automatically collected and centrally stored, typically in Oracle databases. The research division employs at least 300 full-time IT staff, who are spread globally across many sites. In total, the company possesses at least 1,000 full-time IT staff.

The IT staff of the research division are broken out into several functional departments that serve the drug development process:

1. Basic Research IT – This unit supports initial compound discovery;
2. Pre-Clinical IT – This unit supports activities which take place before the compound is tested in humans, such as initial safety assessment testing;

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

3. Clinical IT – This unit supports the clinical testing of compounds in humans;
4. Regulatory IT – This unit supports regulatory compliance, such as the continued monitoring of adverse reactions of drugs that are on the market;
5. Information Integration IT who handles integration between the functional units as well as the IT services that are shared across the division.

Because the research division heavily relies on IT, Information Integration IT originally had a help desk unit and a unit that managed the desktop deployments, maintaining the windows desktop image. Both of these units have been re-organized and merged with similar groups in the Corporate Infrastructure Division.

The Origin of the Environment

The web development unit was originally part of the unit in charge of managing desktop deployments. One of the contractors in the desktop deployment team installed Linux with Apache on a desktop PC to create a web development environment. The contractor wrote a C program that was installed onto all of the divisional desktop units. When a divisional machine booted, the C program would transmit information about the desktop's patch level to the Linux desktop. The information was stored in a MySQL database and could be queried using a cgi binary on the Apache web server. The desktop deployment team used a series of automated scripts to deploy patches to the divisional desktop environment, and the application on the web server allowed the desktop deployment team to track the progress of the patch.

The project was an immediate success within the desktop deployment team, and the contractor was asked to move the web application to a web server in one of the company data centers. The contractor was given a Quad-Processor PIII Compaq Proliant server to build the production web environment. He created the environment from a default build of a Red Hat Linux, with all services enabled. The patch tracking software (and associated database) was moved to the production environment and continued to function on the Proliant server.

Because of the success of the patch tracking software, a web development team was formed to produce more web applications for the desktop support unit. The team was extremely progressive and set up a java-enabled web server, using Tomcat[3] on the production server. They produced an application that rendered XML into HTML and allowed the desktop support teams to track cases submitted to the help desk by using a standard web browser. The original patch tracking software was expanded to allow the user to run queries and create spreadsheets that contained information about the desktop units in the environment. In addition, summary pages showed statistics on patch levels of machines across the global environment.

[3] Tomcat is a java-based web server that was used by the team to serve java server pages and java servlets. See: http://jakarta.apache.org/tomcat/index.html

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

The web development team grew to six full-time employees and five contractors and was separated from the desktop deployment unit. Shortly after the separation, the desktop deployment unit was merged with a similar unit in the Corporate Infrastructure Division. The web development team remained in the research division, but still supports the patch tracking software for the Corporate Infrastructure Division. In addition, their duties were expanded to produce custom java web applications for the research division. They were given a budget to purchase more servers to expand the web environment.

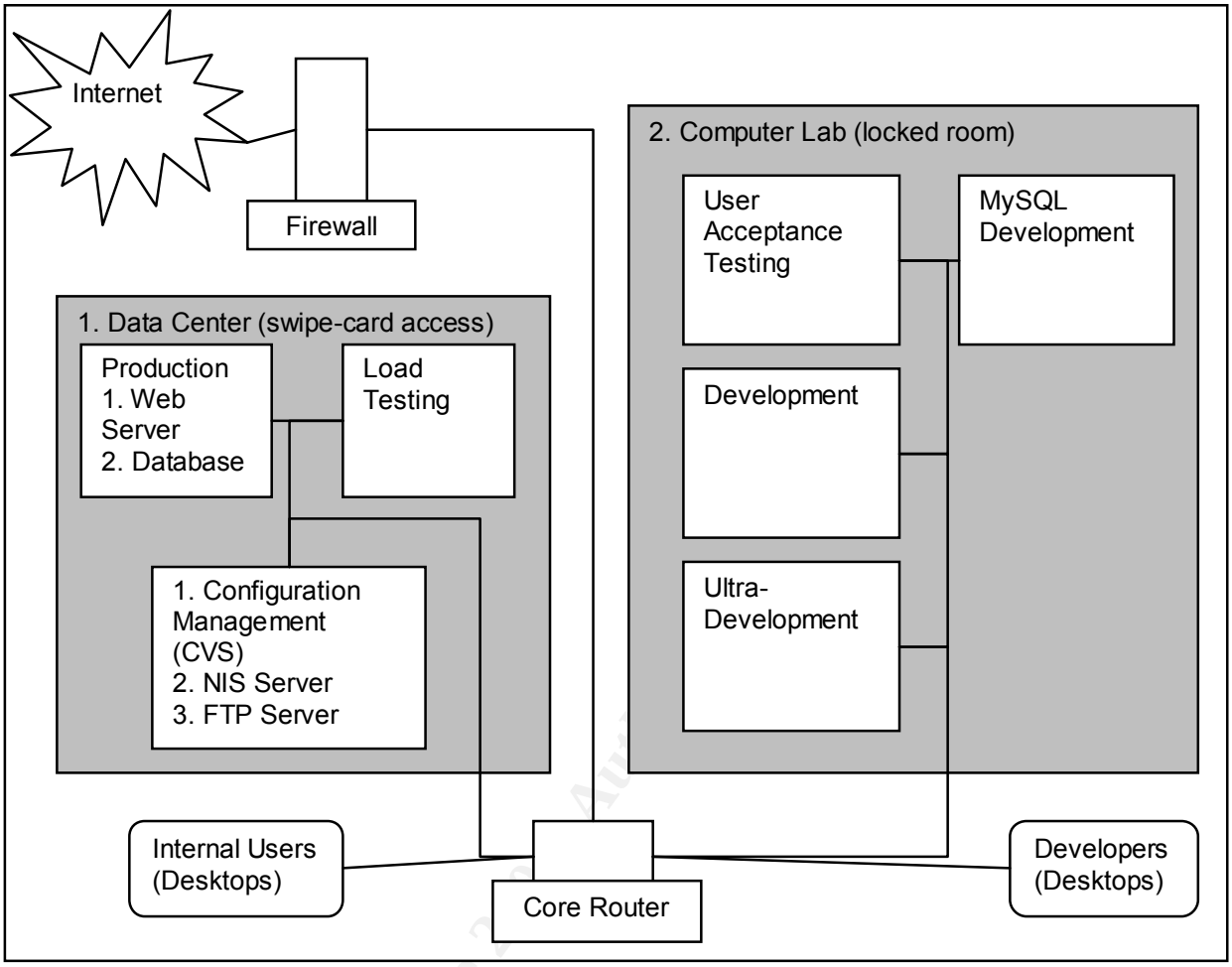
Towards the end of the year, the contractor who had originally created the production environment and patch tracking application left the company with the expiration of his contract. System administration was transferred to a web developer who had shown a proficiency in Linux. At this time, a new contractor was hired to manage all of the team's development projects, and the web development team entered a new phase that followed a systems development life cycle (SDLC).

The Current Environment

The web developer who had been appointed as system administrator was paired with a contractor to create a new environment designed to follow a systems development life cycle (SDLC). The company has issued a formal policy for developers to follow good engineering practices by adopting a formal SDLC. Based on the SDLC practices, the systems administrator builds a new development environment. To configure the new environment, servers and desktop units are purchased. Currently, the environment is physically divided into (1) large servers that are within a data center with swipe-card access control; and (2) development servers that are high-end desktop units and stored in a locked lab area.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Figure 2. Diagram of the Web Server Environment



The entire environment is Linux based and all of the systems are considered internal because they are not exposed to the internet and are protected by the company's main firewall. The servers in the data center are larger server systems with available tape back-up and uninterrupted power supplies (UPS). The data center systems can be described as:

1. Production: this is a quad-processor server that hosts a web server (Apache) with java support (Tomcat) and a database server (MySQL).
2. Load Testing: this is a dual-processor server that hosts a web server (Apache) with java support (Tomcat) and is primarily used to stress-test larger applications. Unlike Production, the Load Testing Server does not host a MySQL database.
3. Configuration Management: this is a dual-processor server that hosts the configuration management system concurrent versions system (CVS). The server also hosts an anonymous FTP server that allows the developers access to team tools. Finally, the server hosts a NIS server that provides user authentication for all of the other systems except for the Departmental Maintenance system in the lab (see lab description).

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

The systems contained within the computer lab are also Linux-based, but are desktop personal computers with increased memory and larger hard drives than the typical desktop. The four main systems are:

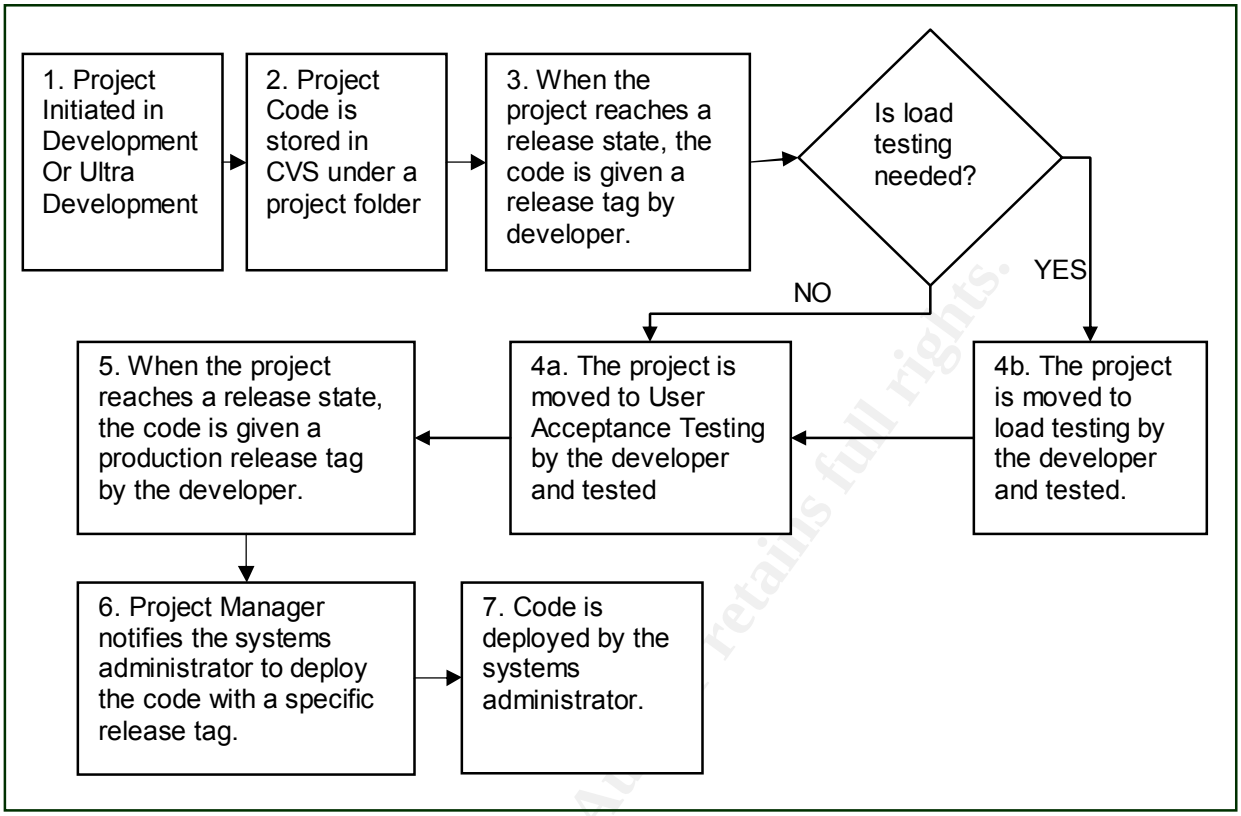
1. Ultra-Development ("Sandbox"): This system hosts applications that may cause the system to be often rebooted during development. Typically developers work on this system if they are working on new source code which may have a number of bugs or is experimental.
2. Development: This system hosts applications that are part of the normal development cycle, such as static web pages and simpler web applications.
3. User Acceptance Testing: As development progresses on a web application, the application is moved to this system where the end user is given an opportunity to test the interface.
4. MySQL Development: This database server supports web applications in initial development. As the application is moved to production, the tables are recreated on the production server.

The environment is managed according to a systems development life cycle in which web applications begin in the development environment and, through a series of reviews, are moved towards production. The Concurrent Versions System (CVS)[4] is critical in the process: in each stage in the lifecycle, the code is tagged with a release tag before moving it to the next stage. The final stage of the cycle is the production release in which the project manager is required to sign-off on the code before it is put in production by the systems administrator. The development cycle is illustrated in Figure 3.

[4] CVS can be found at https://www.cvshome.org/

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Figure 3. Overview of the Systems Development Life Cycle



Current Web Applications and Sites in the Environment

The web servers support a number of sites and applications, including sites with proprietary research and sensitive information. The current sites fall into five major areas:

1. Proprietary Research: Two applications host files that contain summaries of highly-restricted, proprietary research that is routinely accessed by upper management. Access to these sites is currently secured using user names and passwords though Apache .htaccess files[5].
2. Desktop Deployment Information: The environment originated to serve the desktop deployment unit and the production web server hosts an application that is used to track information about the desktop environment. This includes information regarding the software installed on each desktop unit and the patch level. Access to this site is currently secured using user names and passwords though an Apache .htaccess file.
3. Divisional IT Strategy: One web site details the divisional IT strategy, including the technologies and applications being used or tested by the divisional IT staff.

[5] See http://httpd.apache.org/docs/howto/auth.html, which details .htaccess files

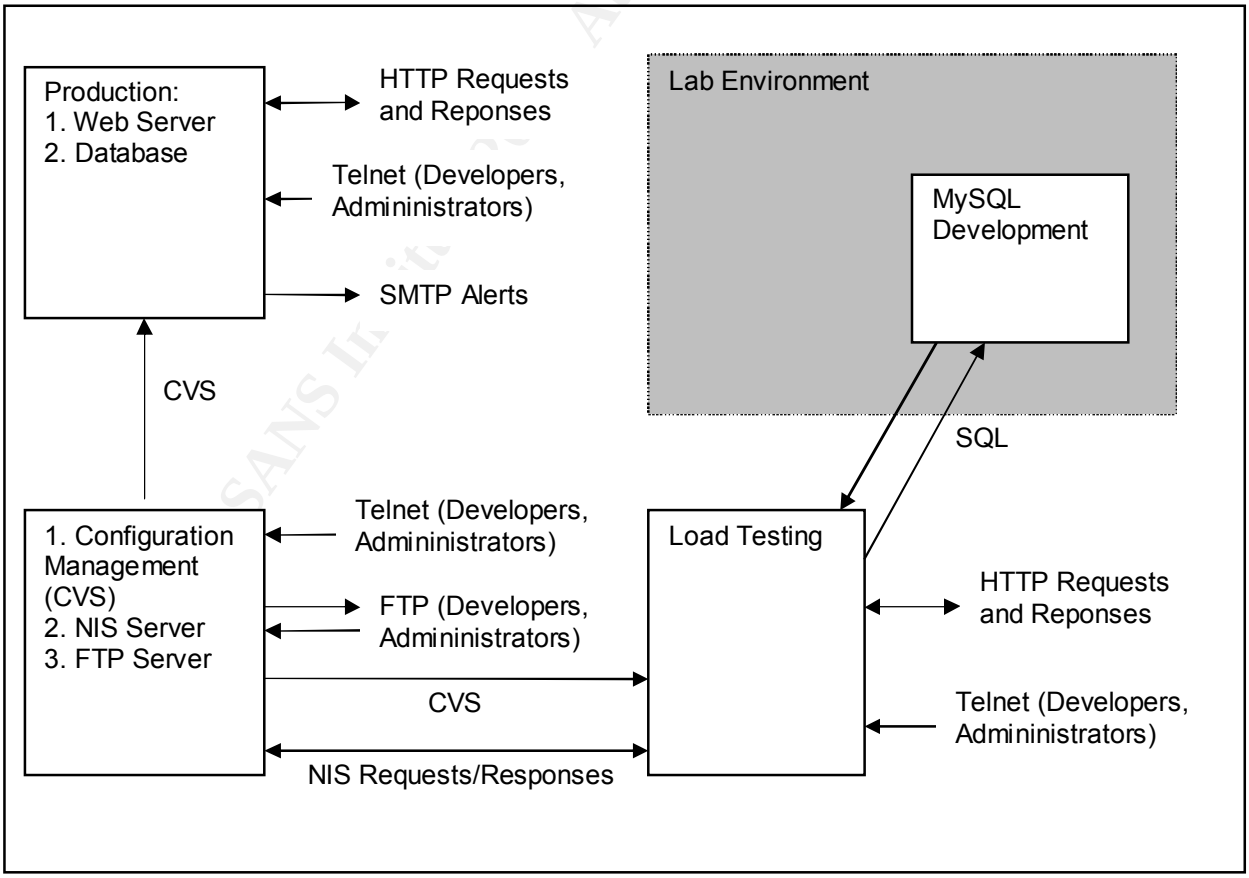
© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

4. Divisional Policy, Training and Procedures: A number of sites have online courses regarding training and divisional procedures. These sites also document divisional policy.
5. Departmental Specific Information: A large number of web sites are dedicated to departmental specific information, including departmental business and organizational charts.

The information flow diagram of the systems in the data center is relatively simple. Internal end-users access the web content on the production server using a browser (HTTP requests). Production applications that draw from a database make SQL calls to an internal database running on the production server. Developers and Administrators access the production server via telnet and content is deployed to production using tagged releases from CVS. In addition, production currently uses sendmail so that it may post alerts to the administrator regarding server performance.

To enable single account administration, all other systems (except production) make calls to the NIS server which also hosts CVS. In addition, the developers use the Configuration Management server to store development tools and FTP is enabled so that users can download the tools. The load testing server also makes calls to test datasets on the MySQL database in the development environment.

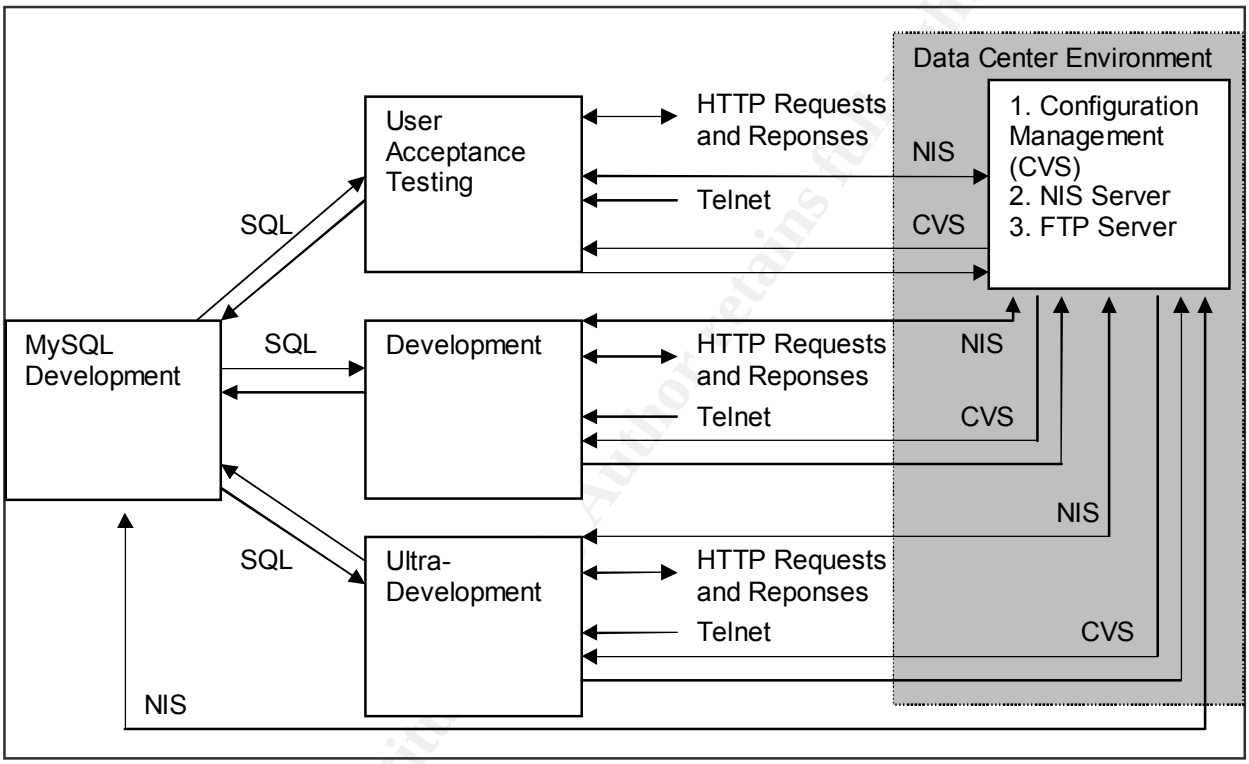
Figure 4. Information Flow in the Data Center Environment



© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

The information flow in the development environment is similar for most of the development machines. Developers and Testers request web pages from the machines. Developers can log into each machine using telnet and move their code to the different machines using CVS for change control. Each machine is configured to use the NIS system for authentication. In addition, many web applications pull test data from the development MySQL database.

Figure 5. Information Flow in the Development Environment



Current State of Security

The internal network of the company is flat. Within the company, several units maintain individual web development environments, which are used to develop applications, such as the environment described. Although a strong firewall policy guards against external threats, internal threats meet with little resistance. This has cost the company in the past when worms have bypassed the firewall, coming in on a mobile user's laptop. Code Red breached the company twice and managed to shutdown several critical web servers, which although they had been patched against the worm, were flooded with HTTP requests resulting in a denial of service. The web development team detected Code Red early in the attack. It identified approximately 20 infected servers across the company and reported the infected machines to the corporate information security unit. Locating the infected servers was difficult because many web developers had

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

established their own environments, which were often in small labs across the company.

Organizationally, a central unit within the corporate infrastructure division manages information security. Their role is documented in the high-level information security policy. The policy defines all information as company property that must be managed to protect the company from losses due to misuse, destruction, fraud or disclosure. Sensitive information resources must possess adequate controls to ensure integrity, confidentiality and availability of the resources.

The information security policy breaks responsibilities down into specific roles that can be summarized as:

1. Corporate Information Security is responsible for implementing company-wide information security policies, standards and guideline. Second, the unit is responsible for administrating a system of information security administrators across the company. Third, the unit is responsible for periodic reviews of the effectiveness information security within the company.
2. Computer Systems Administrators are responsible for implementation and administration of security controls on servers and workstations. Second, they are responsible for monitoring computer resources under their control for security violations and reporting the violations to Central Information Security.
3. Divisional Management is responsible for assigning security administrators and sponsoring security improvements,
4. Divisional Security Administrators are responsible evaluating risks and potential loss values for the information resource. Second, they are responsible for specifying safeguards for the information and determining which users should be allowed access based upon risk and potential loss. Third, they are responsible for maintaining lists of current users of the resource.
5. End-users are responsible to maintain the security of their accounts. End-users are responsible for the use of passwords or other pins/security codes associated with their accounts.

Finally, the high level security policy specifically dictates that compliance to the policy is the responsibility of the user and the individual department. Departments are tasked with planning and implementing necessary security controls to enforce the policy. In addition, they are tasked with monitoring the controls and procedures and may develop standards and procedures to support the high-level policy.

In addition to the high-level security policy, the company also has an information classification policy that is well communicated among all of the employees via an awareness training program. The information classification policy groups information into four categories

1. Public – This information has been released to the public.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

2. Internal Use – This information is considered company property, but is not considered sensitive. All information is considered this category by default. For example, high-level company policies would be considered in this category because they need to be accessible to all employees, but are not publicly released.
3. Restricted – This information is proprietary, personal, or legal in nature and by policy must have controlled access. An example is reports on proprietary chemical structures.
4. Highly Restricted – This information is considered highly sensitive, such as upper management summaries of clinical trial results, where unauthorized disclosure could influence company stock prices.

As part of the document security awareness program, the company includes information on the appropriate handling of restricted documents including clear screen and clear desk procedures. In addition, the company maintains large disposal bins for restricted paper information and has on-site shredding and disposal facilities.

To support the information classification program, the company has implemented a high-level encryption policy that specifies (in summary) that when restricted (or highly restricted) digital documents need to be encrypted when they are removed from their information repository. To facilitate this, the company has deployed a company-wide PKI program with tools that allow the user to encrypt documents and e-mail. The Corporate Information Security group maintains a certificate authority to issue PKI certificates.

The physical security of the company's sites are excellent. The larger sites are essentially gated communities that possess security staff and also onsite emergency staff (paramedics and fire/safety personnel). These sites have multiple security gates that are controlled using swipe-card access and the areas in the site that contain highly sensitive information are isolated and have access control (usually swipe-card readers). The smaller sites typically have a single security person at the main entrance, and also possess swipe-card access.

In the area of personnel security, the company has a rigorous screening program that includes education and background screening. All employees are required to sign a non-disclosure agreement and attend the document security awareness program. Contractors typically go through a reduced screening process, but also must sign a non-disclosure agreement and go through an abbreviated document security training program.

Internally, the company is beginning to tighten the security of its networks. The company maintains anti-virus software on all of its desktop units and scans for e-mail viruses at the mail server. Several months ago, the Corporate Information Security unit hired a security consulting group to audit the internal network, but did not alert the departments that the audit was taking place. The web development team noticed port scans appearing in the server logs as the audit progressed and alerted the Information

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Security unit, but most other departments failed to detect the audit. The Information Security department has shared information regarding the security flaws found in the web production environment.

Scope of Information Security Management System (ISMS)

The upper management has become concerned about the web server environment because of the storage of proprietary information. Recent audit findings found that the system was behind in patches, and because of the recent audit, the web development team is eager to correct the security holes in the environment.

The scope of ISMS is limited to the web server environment and the information resources contained upon the servers. The web server environment includes the Production Web server, the Configuration Management Server, the Load Test Server and all of the development machines in the lab. In addition, the network that hosts these servers is considered in scope, but is considered by the upper management to be well managed and secure.

Although there are other web server environments in the research division, management has placed these out of scope at this time so that they can focus on piloting the creation of an ISMS in the described web server environment. By using the lessons learned from the initial pilot, the research division plans to pull the other environments into scope and also bring them into compliance. Next, the upper management has placed the firewalls out of scope at this time because the web development environment is considered an internal resource (no users will access the servers from the internet). In addition, the firewalls are undergoing similar improvements (based on the audit) that are handled by the Corporate Infrastructure Division.

II. Planning the Implementation of the Information Security Management System (ISMS)

Management Structure

Since the company has a high level security policy in place, the next steps are to create the management committees. Across the division, the Research Information Management Committee (RIMC) is responsible for policies, standards, procedures and guidelines for information management. The RIMC represents the major business areas of the division and is composed of upper-level management. The RIMC manages several areas in information management, including proper document handling and retention. The executive sponsor of the RIMC is the Vice President of the Research Division.

The high-level information security policy defines a role of a divisional security administrator who is responsible for evaluating risks and specifying safeguards (from

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

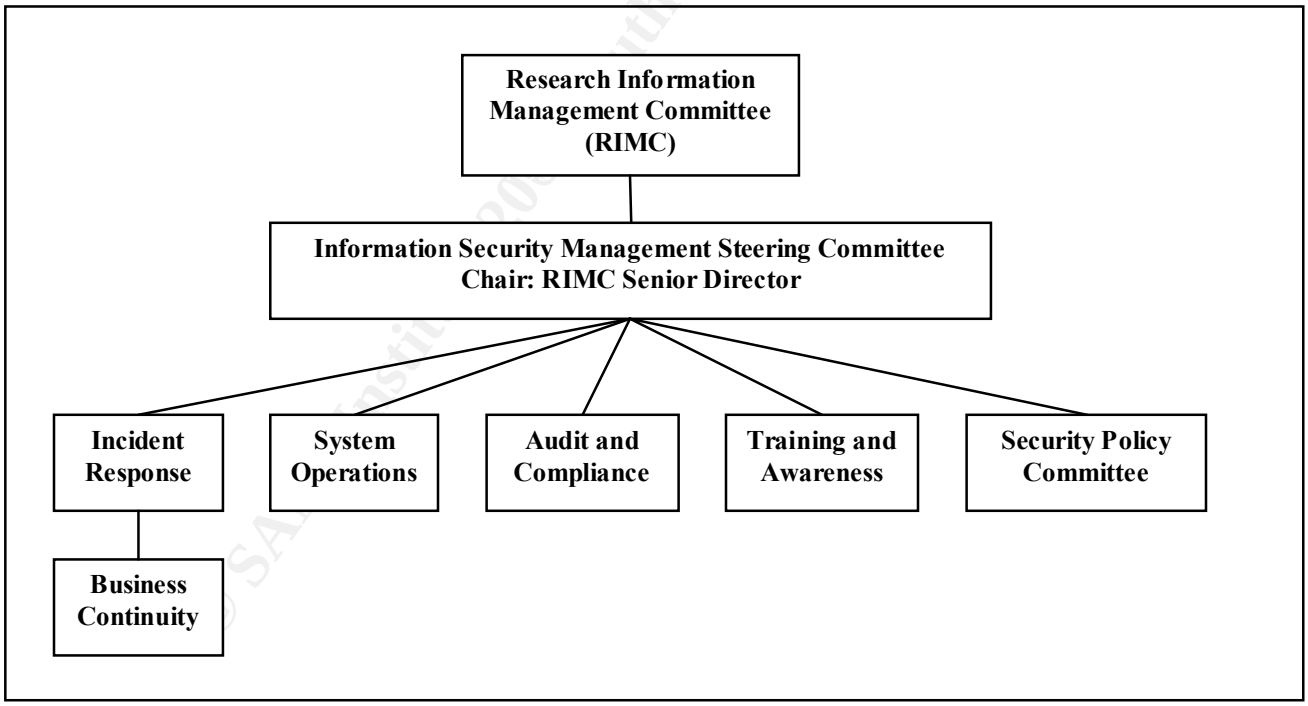
the Current State of Security, above). The RIMC has determined that the divisional security administrator will fill the role of the Information Security Officer (ISO) for the ISMS.

The RIMC has created a sub-committee to act as the Information Security Management Steering Committee. A senior director from the RIMC who reports to the Vice President will chair the committee and the ISO will act as the secretary for this committee, setting agenda items and presenting the current issues. In addition, the steering committee will contain key members from the business and IT who are key leads for the five subcommittees that are defined below.

The mission statement of the Information Security Management Steering Committee is:

The research division recognizes that the internal web internet is an important tool for internal, research communication and the distribution of organizational information. The mission of the ISMS is to secure the information assets contained within the web server environment so that these assets are protected against unauthorized access and modification while preserving the availability of the assets for authorized business users.

Figure 6. Organization of the ISMS committee structure



To carry out its mission, the Information Security Management Steering Committee has chartered five subcommittees:

1. Incident Response: This team is responsible for responding to incidents and directly providing Corporate Information Security with detailed information regarding the incident. This team is also responsible for creating and maintaining

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

an incident handling procedure. The primary members of this team are: (1) a representative from Corporate Information Security; (2) the system administrator; (3) a manager from the development team; and (4) a member of the awareness and training committee who will release official communications regarding incidents; and (5) the information security officer.

   a. Business Continuity Planning is a subcommittee underneath the incident response team. The responsibility of this team is to prepare Business Continuity Procedures and handle communications. This team is composed of: (1) the system administrator; (2) a manager from the development team; (3) a member of the awareness and training committee who will release official communications regarding incidents; and (4) the information security officer.

2. Systems Operations: This team is responsible for operations within the system, including timely patches and updates. This team is responsible for following proper log monitoring procedures and reporting of incidents to the incident response team. This team is composed of (1) the system administrator; (2) a manager from the development team who functions as a second system administrator; (3) a member of the Corporate Network Operations team and (4) the information security officer.

3. Audit and Compliance: This team is responsible for compliance of the system with audit specifications. This committee works directly with the Security Policy and Risk Management Committee to establish audit checklists based upon identified business risks. This team is composed of: (1) a member of the corporate audit team; (2) a senior member of the Security Policy Committee; (3) a member of the Awareness and Training committee; and (4) the information security officer.

4. Awareness and Training: This team is responsible for developing user training and awareness programs regarding system security. This team is composed of: (1) a manager from the development team; (2) a member of the Security Policy Committee (who will help develop awareness of policy decisions; and (3) the information security officer.

5. Security Policy and Risk Management Committee: This team is responsible for developing guidelines or divisional policy to manage the system. This committee is also responsible for asset identification, and identifying risks to the assets that are considered significant to the business. This committee is composed of (1) individuals from the business who are data owners of the web content stored upon the web server; (2) the senior manager of the web development unit; and (3) the information security officer.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Following the establishment of the committees the Information Security Management Steering Committee has established a project plan with deadlines to bring the system closer to ISO 17799 compliance. The project plan can be summarized as:

1. Establish Committees: This step is complete and took one month of meetings.
2. Complete an Asset Inventory: The Security Policy and Risk Management Committee will complete the inventory over a three-week period.
3. Policy Writing: Using the risks identified in the analysis, the Security Policy and Risk Management will begin to draft high level policies to mitigate risk. This must be compete before the implementation phase.
4. Risk Analysis: The Security Policy and Risk Management Committee will use the asset inventory and compete a risk analysis over a period of three weeks to create the risk management plans.
5. Implementation: After the plan for risk management, the committees will work to implement the plans. This will be done over a six-month period.

The Asset Inventory

The first task of the Security Policy and Risk Management Committee was to complete an inventory of assets with criticality classifications. The committee used a series of interviews with the business owners to classify the sensitivity of the content of the information contained on the web sites according the to company's information classification policy. The main information assets fell into these categories:

1. Proprietary Research:
    a. Classification: Highly Restricted
    b. Impact of loss/disclosure: The proprietary research stored on the server is derived from reports from the company's project teams and if lost, could be recreated. Disclosure of the content would reveal the extent and direction of the company's research that could result in a long term loss of competitive advantage (potentially several hundred-thousand dollars to hundreds of millions of dollars[6]).

2. Desktop Deployment Information:
    a. Classification: Restricted
    b. Impact of loss/disclosure: The desktop deployment information stored within the MySQL database contains patch level of desktops in the current environment. If the data was lost, deployment teams would lose a week of work as the information was repopulated into the database. Disclosure would give the attacker excellent information regarding unpatched machines in the environment.

[6] The average cost to develop a drug is $800 million dollars (http://www.phrma.org/publications/publications/brochure/questions/whycostmuch.cfm). Because the sites host information regarding multiple compounds in development, this figure may be in the hundred million dollar range.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

3. Divisional IT Strategy:
    a. Classification: Internal Use
    b. Impact of loss/disclosure: Loss of this data would be an inconvenience until the content was either pulled from CVS or recreated by the business users. Disclosure would give the attacker organizational information that could be useful for social engineering attacks, such as organizational charts.

4. Divisional Policy, Training and Procedures:
    a. Information Asset #4 has the same characteristics as #3.

5. Departmental Specific Information:
    a. Information Asset #5 has the same characteristics as #3.

After classifying the information assets, the Security Policy and Risk Management Committee worked with the Systems Operations Committee and the web development unit to classify other assets, including the hardware that holds the information assets. This analysis is shown in Appendix A – Extended Asset Classification, and uses a series of reports that include: (1) descriptions for each system; (2) the immediate value; (3) potential impacts to confidentiality, integrity and availability; and (4) likely consequences of impacts to confidentiality, integrity or availability. The analysis can be broken into (1) hardware specific, (2) back-up media specific, (3) facility specific, and (4) personnel specific. A summary of the asset inventory in Appendix A is prepared below:

(1) Hardware

1. Production Apache/Tomcat Web Server and Database Server:
    a. Classification: Highly Restricted
    b. Summary of impact of loss/disclosure: The immediate value of the server is approximately $51,000. Disclosure of the proprietary content would reveal the extent and direction of the company's research that could result in a long term loss of competitive advantage (potentially hundreds of thousands of dollars to hundreds of millions of dollars). Disclosure of the desktop deployment information would give the attacker excellent information regarding unpatched machines in the environment.

2. Production CVS/NIS Server:
    a. Classification: Restricted
    b. Impact of loss/disclosure: The immediate value of the server is approximately $51,000. CVS does not store any of the proprietary content from production, but disclosure of the other departmental web sites could give the attacker organizational information that could be useful for social engineering attacks. An attacker who gains root on this server would be able to use the NIS system to compromise the development and Load Test Environments.

3. Load Test Apache/Tomcat Web Server:

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

a. Classification: Internal Use – Access restricted to web development unit
b. Impact of loss/disclosure: The immediate value of the server is approximately $51,000. Disclosure of development code on this server may reveal flaws that could be used to compromise the production environment.

4. Development Servers:
    a. Classification: Internal Use – Access restricted to web development unit
    b. Impact of loss/disclosure: The immediate value of a PC is approximately $3,200. Disclosure of development code on these servers may reveal flaws that could be used to compromise the production environment. Some content may contain organizational information that could be useful for social engineering attacks.

5. Network and Switches (and the data that travels across them):
    a. Classification: Restricted
    b. Impact of loss/disclosure: The immediate value of the network is estimated at $ 500,000 for the site. The management is mainly concerned about the information traveling across the network, because it may include usernames and passwords.

(2) Back-up Media

1. Tape for the Production Apache/Tomcat Web Server and Database Server
    a. Classification: Highly Restricted
    b. Summary of impact of loss/disclosure: The tape includes back-ups of all of the production content and the MySQL database. Disclosure of the proprietary content would reveal the extent and direction of the company's research that could result in a long term loss of competitive advantage (potentially hundreds of thousands of dollars to hundreds of millions of dollars). Disclosure of the desktop deployment information in the database would give the attacker excellent information regarding unpatched machines in the environment.

2. Tape for the Production CVS/NIS Server:
    a. Classification: Restricted
    b. Impact of loss/disclosure: The tape contains back-ups of CVS content only. Disclosure of the departmental web site content could give the attacker organizational information that could be useful for social engineering attacks.

(3) Facilities

1. Data Center:
    a. Classification: Highly Restricted
    b. Impact of loss/disclosure: The data center contains the production servers of the web development team as well as the production servers for other teams in the company. The immediate value is approximately $ 20 – 25

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

million dollars in servers and hardware. Attackers with physical access to the data center could gain access to proprietary information, corrupt production data or disrupt normal business operations by destroying hardware, causing hundreds of thousands of dollars in damage.

2. Development Lab:
    a. Classification: Restricted – Access Controlled.
    b. Impact of loss/disclosure: The development lab contains only the development servers of the web development team. The immediate value is $ 20,000 in servers and hardware. Attackers with physical access could gain access to development code and departmental web site content that could give the attacker organizational information that could be useful for social engineering attacks.

(4) Personnel

1. Managers and the System Administrator:
    a. The managers and system administrator have root access to all servers and can see or modify any content.
    b. One of the managers fills in for the system administrator when he is on leave, but deployments slow down during this period.
2. Developers and the Tester:
    a. The developers and tester have root access to all development servers and can see or modify any content. The developers and tester have user access to all production servers.
    b. The developers fills in for the tester when he is on leave, and testing may slow down during this period.
3. Contractors:
    a. The contractors have root access to all development servers and can see or modify any content. The developers and tester have user access to all production servers.

Policies

The next task of the Security Policy and Risk Management Committee was to identify high level policies that were needed to manage risks in the ISMS. A review of ISO 17799 revealed two policies that were needed by the system. First, the ISMS needed an access control policy (section 9.1 of the ISO 17799 standard) that directs the creation of standards for (1) user registration and de-registration; (2) allocation of system privileges; (3) standards for handling and use of user credentials and (4) the periodic review of access rights. This policy was drafted by the Security Policy and Risk Management Committee. Second, the ISMS needed a policy for Business Continuity planning (section 11.1 of the ISO 17799 standard). Currently, the company uses standards for business continuity planning, but did not have a formal business continuity policy. The policy was drafted by the Business Continuity Planning Committee.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

As the risk management process progressed, the team identified a third policy, Security Engineering in the Systems Development Life Cycle, which would aid in managing security risks identified in the development cycle (sections 10.1 and 10.2 of the ISO 17799 standard). The Systems Operations committee drafted this policy.

The policies were presented to the ISMS committee and were approved at a divisional level by the Research Information Management Committee. Although the policies were initially proposed and ratified at a divisional level, they were ultimately approved by the Corporate Information Management Committee to be company-wide, because of their scope.

In addition to the policies, several standards and procedures were identified as being needed for implementation. These procedures and standards will be presented in the section Plans for Risk Management. The policies are outlined below and presented in greater detail in Appendix B – Policies.

1. Policy Name: System and Application Access Control
    a. Purpose: To protect systems and applications from unauthorized use and to protect the confidentiality of sensitive information.
    b. Audience: This policy covers all company information systems and applications. This policy applies to all staff, including developers, who will implement the access control and data owners who will specify the level of control.
    c. Section 9.1 of the ISO 17799 standard, including 9.1.1 (Access Control policy), 9.2.1 - 9.2.4 (User Access Management).

2. Policy Name: Business Continuity Planning
    a. Purpose: The purpose of this policy is to provide for planning that assures the continuation of business operations in the event of a disruption or a disaster.
    b. Audience: This policy applies to systems and assets that are considered essential for business operations. Data owners are responsible for coordinating with system administrators to produce and implement plans for business continuity for the essential systems.
    c. Section 11.1 (11.1.1-11.1.5) of the ISO 17799 standard (Aspects of Business Continuity Management).

3. Policy Name: Security Engineering in the Systems Development Life Cycle
    a. Purpose: To establish guidance on security engineering in the Systems Development Life Cycle.
    b. Audience: This policy will apply to all information systems that are governed by the Systems Development Life Cycle. This policy applies to all development teams, including data owners who are responsible for proper data classification and communicating the risks to the development team so that controls may be implemented to mitigate the risks.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

c. Section 10.1 of the ISO 17799 standard (Security requirements of systems).

Risk Identification and Analysis Process

After asset identification, the Security Policy and Risk Management Committee began a series of interviews with the web development unit and the business end-users to determine past faults that had been observed in the ISMS. As the team began to conduct their interviews, they discovered that the system administrator had been tracking the frequency of many system faults to improve the operations. Using his notes and the interview data, the committee identified several high level failures:

1. Application code fails in production (resulting in loss of available of application)
2. The production server is offline or unavailable.
3. The NIS/CVS server is offline or unavailable.
4. Production code is modified without permission.
5. Development servers are offline or unavailable.
6. Sensitive Information is disclosed. (This has never been detected, but the ISMS steering committee has asked the team to consider this possibility)

The risk analysis approach that the team used was Cause Consequence Analysis (CCA). The high level faults were used to construct fault trees to gain a deeper understanding of the factors that went into the faults. As the analysis progressed, faults that were related to one another were linked together, such as an application failing because the production code had been modified without permission. In addition, all of the server offline or unavailable faults were combined as a general fault tree because the generic faults were similar for most servers. The three major trees are found in Appendix C – Fault Tree Analysis.

Using the fault trees, with the notes of the system administrator, the committee flagged several faults that were seen as important to the operations of the ISMS. In the first two Fault Trees (Server Unavailable and Application Code Fails in Production) the group flagged events that had occurred in the past, and these faults are illustrated as heavy black boxes in trees one and two.

In the third fault tree, the Security Policy and Risk Management Committee flagged events that had been reported in the previous security audit (passwords unencrypted and patches not up to date) or were known to occur in the company at large (no access control policy, data not classified properly, and back-up media not properly secured).

Using the fault trees, the Security Policy and Risk Management Committee pulled together a series of flash cards that had information on the major faults that had been flagged in each fault tree (the bold boxes). The flash cards are shown in Appendix D. Using the flash cards and the asset inventory as a guide to the severity of the fault, the

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

team combined some events together and separated other events according to asset values. This allowed the team to prioritize the events in order of severity. For example, a compromise by a hacker on the production system was considered to be a high severity and was a separate event (event #6 in Appendix D) when compared to the same event in development (event #12 in Appendix D).

For each event flashcard, the committee first created a concise description of the event that linked the boxes of the fault tree together, such as "Disclosure of sensitive information due to improper data classification", which was derived from the boxes: (1) Sensitive information stored on the production server is compromised; (2) Data was unprotected; and (3) Data owner did not classify data properly.

Next, the team created a description of the event and listed the frequency of this event that had been derived from interviews with the web development unit and the business end-users. Third, the committee listed any known preventive controls for the event. Finally, the team constructed a series of event trees to examine detective and corrective controls in the environment. Through interviews with the web development unit, they found that the team had a series of informal detective controls that could be incorporated into documented operating procedures (ISO 17799 section 8.1.1). For example, the systems administrator had been highly successful at detecting network worms because the first thing he did in the morning was to scan over some of the systems logs looking for abnormal traffic.

Using the event flash cards, the Security Policy and Risk Management Committee put together a high-level plan for risk management that is shown in Appendix E. The committee ranked the events in priority based on severity and likelihood. For each event, the team examined which controls were already in place, and which controls would be needed. Using the high-level plan and the asset inventory, the Security Policy and Risk Management Committee presented a review to the Information Security Management Steering Committee for approval.

One interesting discussion point in the review was that the web development team typically focused on system availability[7] (such as loss of availability due to improper testing), but the ISMS committee was more concerned with data confidentiality, which has a higher loss value. This point was relayed to the Awareness and Training committee to educate the developers on the overall risks to the system. In the end, the ISMS committee reviewed and approved the high-level plan based upon the asset inventory.

Plans for Risk Management

Using the high-level plan (Appendix E), the Security Policy and Risk Management Committee created a report (Table 1. Plan for Risk Management) with

[7] This is not surprising. Most of the complaints the team hears are when the system is unavailable.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

recommendations of controls to mitigate specific risks. For some of the risks, the team considered that the existing controls were adequate. These risks were:

1. Attacker gains physical access to production server in data center
2. Attacker gains physical access to development server in development laboratory
3. Information in production MySQL databases corrupted
4. Development server unavailable due to missing network cable

For the implementation plan, the team recommended that several risks should be grouped together, because they were similar in nature:

1. All of the risks under “server compromised by worm” are grouped together, because the development environment would be used to test patches and software as part of the patch release cycle.
2. Similarly, all of the “server compromised by hacker (using network)” are grouped together because patches would be tested first in development before being put into production.
3. Although contractors are considered more of a risk to information loss, the mitigation strategy for “user abuses privileges and steals sensitive information” was similar and the contractor and employee controls are written as a group.
4. All of the denial of service attacks needed incident handling procedures to respond to alerts and were combined into a single category of DOS attacks.

In addition, the committee identified several specific standards and procedures that would need to be implemented as controls for the system:

Procedures:
1. Procedures for responding to security incidents (ISO 17799:8.1.3).
2. Procedures for Information Back-up (ISO 17799:8.4.1).
3. Procedures for disaster recovery and business continuity (ISO 17799:11.1).
4. Standard Operating Procedures for management of the production and development environment. (ISO 17799:8.1.1).
5. Change Control Procedures (ISO 17799:10.5.1).

Standards:
1. standards for formal user registration and de-registration (ISO 17799:9.2.1).
2. standards for allocation of system privileges (ISO 17799:9.2.2).
3. standards for handling and use of user credentials, such as passwords (ISO 17799:9.2.3).
4. standards for the periodic review of access rights (ISO 17799:9.2.4).
5. standards for network controls (ISO 17799:8.5.1).
6. standards for Systems Development and maintenance (ISO 17799:10.1 and 10.2).
7. Encryption standards (for the Encryption Policy, ISO 17799:10.3.1).

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Table 1. Plan for Risk Management

1. Nature of the Threat: **Disclosure of sensitive information due to improper data classification (data is unprotected)**	ISO 17799 Sections: 5.2.1 - 5.2.2 and 10.1.1
Vulnerability: Data owners sometimes fail to classify their data, leaving it unprotected on the server.	
Likelihood: 1 – 2 times per year, sensitive information has been found to be unprotected on other servers in the company.	
Risk Level: High - on the Production Web Server, loss of proprietary information could cost the company potentially hundreds of thousands of dollars to hundreds of millions of dollars.	
Description of Controls: Mitigate the risk by (1) setting up checkpoints in development to establish controls for sensitive information and (2) establish a periodic audit of web site content.	
Reasons for selecting controls: The checkpoints increase the likelihood that this issue will be detected in development. The audits ensure that content is compliant.	
Risk Level after implementing control: Low	

2. Nature of the Threat: **Sensitive information stolen from back-up media**	ISO 17799 Sections: 10.3.2
Vulnerability: Information on back-up tapes is currently unencrypted.	
Likelihood: Unknown - Currently, tapes are stored in a locked back room of the data center that is watched by tape operators.	
Risk Level: High - on the Production Web Server, loss of proprietary information could cost the company potentially hundreds of thousands of dollars to hundreds of millions of dollars.	
Description of Controls: Mitigate the risk by (1) establishing standards for data encryption on back-up media and (2) implementing encryption standards.	
Reasons for selecting controls: These controls will make back-up information useless if stolen.	
Risk Level after implementing control: Very Low	

3. Nature of the Threat: **Authorized user abuses privileges and steals sensitive information**	ISO 17799 Sections: 9.2
Vulnerability: The system has no access control procedures.	
Likelihood: Employees go through a rigorous screening process and sign a non-disclosure agreement.	
Risk Level: High - loss of proprietary information could cost the company potentially hundreds of thousands of dollars to hundreds of millions of dollars.	
Description of Controls: Mitigate the risk by (1) establishing standards for informing the user of proper use guidelines; (2) promote periodic review of access privileges; and (3) establish audit logs for user access.	
Reasons for selecting controls: These standards re-enforce proper use of the system and monitor for transgressions.	
Risk Level after implementing control: Medium to Low (Insider attacks will still have a high severity).	

4. Nature of the Threat: **Contractor abuses system privileges and steals sensitive information**	ISO 17799 Sections: 9.2
Vulnerability: The system has no access control procedures.	
Likelihood: Contractors have to sign a non-disclosure agreement. Trends in Proprietary Information Loss (2002)[8] rate them a higher risk to information loss than employees.	

[8] Trends in Proprietary Information Loss – Survey Report, September 2002; Sponsored by Price Waterhouse Coopers, U.S. Chamber of Commerce and ASIS Foundation.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Risk Level: High - loss of proprietary information could cost the company potentially hundreds of thousands of dollars to hundreds of millions of dollars.
Description of Controls: Mitigate the risk by (1) establishing standards for informing the user of proper use guidelines; (2) promote periodic review of access privileges; and (3) establish audit logs for user access.
Reasons for selecting controls: These standards re-enforce proper use of the system and monitor for transgressions.
Risk Level after implementing control: Medium to Low (Insider attacks will still have a high severity).

5. Nature of the Threat: **Packet Sniffing on internal network**	ISO 17799 Sections: 8.5.1 and 10.3.2
Vulnerability: Communication channels are currently unencrypted, such as the use of telnet.	
Likelihood: Unknown - The network currently uses switches to route information and makes sniffing slightly more difficult (but not impossible).	
Risk Level: High - since many channels are unencrypted, the attacker would easily collect usernames and passwords by packet sniffing.	
Description of Controls: Mitigate the risk by (1) establishing encryption standards for communication channels and (2) establish encrypted communication channels	
Reasons for selecting controls: These controls encrypt the communication channels.	
Risk Level after implementing control: Very Low	

6. Nature of the Threat: **Production Web Server compromised by internal hacker (using network)**	ISO 17799 Sections: 8.1.3, 9.5, 9.7.2,10.4.1 and 11.1.3
Vulnerability: The server is behind in patches.	
Likelihood: Unknown - during an audit, the development servers were compromised.	
Risk Level: High - loss of proprietary information could cost the company potentially hundreds of thousands of dollars to hundreds of millions of dollars.	
Description of Controls: Mitigate the risk by (1) establishing incident handling procedures; (2) establishing business continuity plans (3) establishing a regular patch cycle; (4) installing integrity checking software; and (5) installing an intrusion detection system.	
Reasons for selecting controls: These controls patch the system and put in controls that will detect a compromise.	
Risk Level after implementing control: Low	

7. Nature of the Threat: **CVS/NIS Server compromised by hacker (using network)**	ISO 17799 Sections: 8.1.3, 9.5, 9.7.2,10.4.1 and 11.1.3
Vulnerability: The server is behind in patches.	
Likelihood: Unknown - during an audit, the development servers were compromised.	
Risk Level: High - the attacker may be able to compromise NIS or the CVS source code and could impact the entire environment.	
Description of Controls: Mitigate the risk by (1) establishing incident handling procedures; (2) establishing business continuity plans (3) establishing a regular patch cycle; (4) installing integrity checking software; and (5) installing an intrusion detection system.	
Reasons for selecting controls: These controls patch the system and put in controls that will detect a compromise.	
Risk Level after implementing control: Low	

8. Nature of the Threat: **CVS/NIS or Production Web Server cannot be recovered after a disaster**	ISO 17799 Sections: 8.1.3, 11.1.3 and 8.6.1
Vulnerability: The system has no disaster recovery plans	

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Likelihood: Unknown, these servers have never needed to be restored. Currently, these servers are backed-up once per week.
Risk Level: High - without procedures, the server would take some time to recover
Description of Controls: Mitigate the risk by (1) establishing incident handling procedures; (2) establishing business continuity plans; and (3) establishing back-up media handling procedures, including off-site storage.
Reasons for selecting controls: These controls establish procedures for disaster recovery.
Risk Level after implementing control: Low

9. Nature of the Threat: **Production Web Server compromised by worm**	ISO 17799 Sections: 8.1.3, 8.3, and 11.1.3
Vulnerability: The server is behind in patches.	
Likelihood: 1 - 2 times per year	
Risk Level: Medium - network worms are quickly detected by the network operations unit, so the infection would be contained. Still the server might need to be restored and production code and data may be lost.	
Description of Controls: Mitigate the risk by (1) establishing incident handling procedures; (2) establishing business continuity plans (3) establishing a regular patch cycle; (4) installing anti-virus software on the server; and (5) installing an intrusion detection system.	
Reasons for selecting controls: These controls establish procedures for system patches and install software to detect and control infections.	
Risk Level after implementing control: Low	

10. Nature of the Threat: **CVS/NIS Server Compromised by worm**	ISO 17799 Sections: 8.1.3, 8.3, and 11.1.3
Vulnerability: The server is behind in patches.	
Likelihood: 1 - 2 times per year	
Risk Level: Medium - network worms are quickly detected by the network operations unit, so the infection would be contained. Still the server might need to be restored and development code could be lost.	
Description of Controls: Mitigate the risk by (1) establishing incident handling procedures; (2) establishing business continuity plans (3) establishing a regular patch cycle; (4) installing anti-virus software on the server; and (5) installing an intrusion detection system.	
Reasons for selecting controls: These controls establish procedures for system patches and install software to detect and control infections.	
Risk Level after implementing control: Low	

11. Nature of the Threat: **Load Test Server compromised by hacker (using network)**	ISO 17799 Sections: 8.1.3, 9.5, 9.7.2,10.4.1 and 11.1.3
Vulnerability: The server is behind in patches.	
Likelihood: Unknown - the server is behind in patches and during an audit, the development servers were compromised.	
Risk Level: Medium - compromise would not lead to immediate losses, but could be used to gain a foothold in the environment.	
Description of Controls: Mitigate the risk by (1) establishing incident handling procedures; (2) establishing business continuity plans (3) establishing a regular patch cycle; (4) installing integrity checking software; and (5) installing an intrusion detection system.	
Reasons for selecting controls: These controls establish procedures for system patches and install software to detect a system compromise.	
Risk Level after implementing control: Low	

12. Nature of the Threat: **Development Server compromised by hacker (using**	ISO 17799 Sections: 8.1.3, 9.5, 9.7.2,10.4.1 and 11.1.3

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

network)	
Vulnerability: The server is behind in patches.	
Likelihood: Unknown - the development servers are behind in patches and during an audit, the development servers were compromised.	
Risk Level: Medium - compromise would not lead to immediate losses, but could be used to gain a foothold in the environment.	
Description of Controls: Mitigate the risk by (1) establishing incident handling procedures; (2) establishing business continuity plans (3) establishing a regular patch cycle; (4) installing integrity checking software; and (5) installing an intrusion detection system.	
Reasons for selecting controls: These controls establish procedures for system patches and install software to detect a system compromise.	
Risk Level after implementing control: Low	

13. Nature of the Threat: **Information in Production MySQL Database compromised by hacker**	ISO 17799 Sections: 8.1.3, 9.5, 9.7.2,10.4.1 and 11.1.3
Vulnerability: The production server is behind in patches.	
Likelihood: Unknown, but the production server is behind in patches.	
Risk Level: Medium - compromise would give the attacker information on vulnerable machines in the environment.	
Description of Controls: Mitigate the risk by (1) establishing incident handling procedures; (2) establishing business continuity plans (3) establishing a regular patch cycle; and (4) installing an intrusion detection system.	
Reasons for selecting controls: These controls establish procedures for system patches and install software to detect a system compromise.	
Risk Level after implementing control: Low.	

14. Nature of the Threat: **Unauthorized modification of production code by developer**	ISO 17799 Sections: 9.2.4 and 10.5.1-10.5.2
Vulnerability: Some privileges are set incorrectly in production allowing developers write access to the code.	
Likelihood: 2 times per year	
Risk Level: Medium - past incidents led to the breakdown of the application, but are indicative that procedures need to be established to prevent developers from modifying production code.	
Description of Controls: Mitigate the risk by (1) establishing development guidelines; (2) periodic audits of user privileges and eliminate unnecessary access rights; and (3) installing integrity checking software.	
Reasons for selecting controls: The guidelines will inform developers of the proper procedures and the integrity checking software will monitor that the guidelines are followed.	
Risk Level after implementing control: Low	

15. Nature of the Threat: **Developer accidentally deletes code directly on the CVS repository.**	ISO 17799 Sections: 10.4.3 and 11.1.3
Vulnerability: The developers have write access to the CVS repository that contains the development code.	
Likelihood: Has never occurred	
Risk Level: Medium - disruption of integrity could set development of applications back several weeks.	
Description of Controls: Mitigate the risk by (1) establishing development procedures and (2) establishing business continuity plans.	
Reasons for selecting controls: These controls establish procedures for the proper use of CVS and recovery plans if the procedures fail.	

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Risk Level after implementing control: Low

16. Nature of the Threat: **Network switch unavailable due to unintentional DOS (application error)**	ISO 17799 Sections: 8.1.3 and 8.5.1
Vulnerability: Heavy internal traffic may disable a network switch.	
Likelihood: 1 time in 3 years	
Risk Level: Medium - the production systems were unavailable during this period.	
Description of Controls: Mitigate the risk by establishing incident handling procedures in response to network monitoring alerts	
Reasons for selecting controls: Internal DOS is difficult to prevent, but the procedures will reduce the impact.	
Risk Level after implementing control: Medium to Low (the DOS is not prevented from occurring).	

17. Nature of the Threat: **Attacker gains physical access to production server in data center**	ISO 17799 Sections: 7.1
Vulnerability: Physical access would allow the attacker to compromise servers or disrupt business operations.	
Likelihood: unknown - The data center is in a separate building and is manned by personnel 24 x 7. The doors are locked with swipe card access and visitors must sign in. The physical controls to prevent and protect this are currently excellent.	
Risk Level: Low - this risk is well controlled.	
Description of Controls: Accept the current risk - The management considers the current controls adequate.	
Reasons for selecting controls: Not Applicable	
Risk Level after implementing control: Low	

18. Nature of the Threat: **Attacker gains physical access to development server in development laboratory**	ISO 17799 Sections: 7.1
Vulnerability: Physical Access would allow the attacker access to development servers that do not contain proprietary information. The attacker could find information useful for social engineering, but would be have to hunt through the servers for data.	
Likelihood: unknown - The development lab is a key-locked room and is unlocked only when one of the development team is using the room	
Risk Level: Low - this risk is controlled and the impact is fairly low.	
Description of Controls: Accept the current risk - The management considers the current controls adequate.	
Reasons for selecting controls: Not Applicable	
Risk Level after implementing control: Low	

19. Nature of the Threat: **Load Test Server Compromised by worm**	ISO 17799 Sections: 8.1.3, 8.3, and 11.1.3
Vulnerability: The server is behind in patches.	
Likelihood: 1 - 2 times per year	
Risk Level: Low - network worms are quickly detected by the network operations unit, so the infection would be contained. The server might need to be restored but has no production content stored on it making the impact low.	
Description of Controls: Mitigate the risk in production and development servers by (1) establishing incident handling procedures; (2) establishing business continuity plans (3) establishing a regular patch cycle; (4) installing anti-virus software on the server; and (5) installing an intrusion detection system.	

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Reasons for selecting controls: These controls establish procedures for system patches and install software to detect and control infections.
Risk Level after implementing control: Very Low

20. Nature of the Threat: **Development Server compromised by worm**	ISO 17799 Sections: 8.1.3, 8.3, and 11.1.3
Vulnerability: The server is behind in patches.	
Likelihood: 1 - 2 times per year	
Risk Level: Low - network worms are quickly detected by the network operations unit, so the infection would be contained. The server might need to be restored but has no production content stored on it making the impact low.	
Description of Controls: Mitigate the risk in production and development servers by (1) establishing incident handling procedures; (2) establishing business continuity plans (3) establishing a regular patch cycle; (4) installing anti-virus software on the server; and (5) installing an intrusion detection system.	
Reasons for selecting controls: These controls establish procedures for system patches and install software to detect and control infections.	
Risk Level after implementing control: Very Low	

21. Nature of the Threat: **Information in production MySQL databases corrupted**	ISO 17799 Sections: 10.2.2
Vulnerability: In the past, the patch tracking software miss-writes to the database, corrupting it.	
Likelihood: 1 - 2 times per year	
Risk Level: Low - the system admin repaired the database and, over time, the data was repopulated from the desktop units reporting their data. There was little impact.	
Description of Controls: Accept the current risk - The database repopulated rapidly in past incidents.	
Reasons for selecting controls: Not Applicable.	
Risk Level after implementing control: Low	

22. Nature of the Threat: **Server unavailable due to DOS attack (worm)**	ISO 17799 Sections: 6.3.1 and 8.1.3
Vulnerability: Internet worms that penetrate the firewall sometimes cause inadvertent denial of service attacks on internal servers.	
Likelihood: 1 - 2 times per year	
Risk Level: Low - network worms are quickly detected by the network operations unit, so the infection would be contained resulting in a low impact.	
Description of Controls: Mitigate the risk by establishing incident handling procedures to report the incident rapidly.	
Reasons for selecting controls: The internal DOS is difficult to prevent, but the procedures will reduce the impact.	
Risk Level after implementing control: Very Low	

23. Nature of the Threat: **Server unavailable due to unintentional DOS (application error)**	ISO 17799 Sections: 6.3.1 and 8.1.3
Vulnerability: The patch tracking software is sometimes overwhelmed by data and spawns enough processes to use up most of the system resources,	
Likelihood: 5 times per year	
Risk Level: Low - customers complained, but the situation was rapidly detected by internal scripts and corrected.	
Description of Controls: Mitigate the risk by establishing incident handling procedures in response to internal script alert.	

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Reasons for selecting controls: The internal DOS is difficult to prevent, but the procedures will reduce the impact. In addition, the internal alert may detect the error before the problem overwhelms the server.
Risk Level after implementing control: Very Low

24. Nature of the Threat: **Loss of application availability due improper testing**	ISO 17799 Sections: 10.5.1 - 10.5.2
Vulnerability: Developers sometimes cut corners and do not test the application properly.	
Likelihood: 3 times per year	
Risk Level: Low - this resulted in a single broken application.	
Description of Controls: Mitigate the risk by establishing development procedures.	
Reasons for selecting controls: The controls will re-enforce proper development procedures.	
Risk Level after implementing control: Very Low	

25. Nature of the Threat: **Production Web Server unavailable due accidental shut-down**	ISO 17799 Sections: 8.1.1 and 8.1.3
Vulnerability: The administrator may accidentally shutdown the production server.	
Likelihood: 1 time per year	
Risk Level: Low - this resulted in a short period of unavailability until the server was rebooted.	
Description of Controls: Mitigate the risk by establishing incident handling procedures.	
Reasons for selecting controls: The controls will re-enforce proper system operations.	
Risk Level after implementing control: Very Low	

26. Nature of the Threat: **Application server unavailable due to misconfiguration**	ISO 17799 Sections: 10.5.1 - 10.5.2
Vulnerability: The system administrator makes error in writing a web server configuration file.	
Likelihood: 4 times per year	
Risk Level: Low - this resulted in a short period of unavailability until the system administrator rolled back the changes.	
Description of Controls: Mitigate the risk by establishing formal change control procedures.	
Reasons for selecting controls: The controls will re-enforce proper system operations.	
Risk Level after implementing control: Very Low	

27. Nature of the Threat: **Development server unavailable due to missing network cable**	ISO 17799 Sections: 7.2.3
Vulnerability: The network cables may be removed by individuals within the lab area.	
Likelihood: 3 to 4 times per year	
Risk Level: Low - the developers detected it and replaced the cable.	
Description of Controls: Accept the risk - This has low impact	
Reasons for selecting controls: Not Applicable	
Risk Level after implementing control: Low	

28. Nature of the Threat: **Incorrect Code Deployed or Code Deployed Incorrectly, resulting in code that does not operate correctly in production.**	ISO 17799 Sections: 10.5.1 - 10.5.2
Vulnerability: Sometimes the system administrator forgets the deployment instructions. Sometimes, the development team fails to pass the administrator the instructions.	
Likelihood: 5 to 6 times per year	
Risk Level: Low - the result was a broken application	
Description of Controls: Currently there are informal review processes to check the code after	

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

deployment. The risk can be mitigated by formalizing the procedures.
Reasons for selecting controls: The procedures enforce proper deployments.
Risk Level after implementing control: Very Low

III. Implementation (the "Do" phase)

Correcting the Problems Identified in the Risk Management Plan

Using the plan for risk management, the Security Policy and Risk Management Committee drafted a report that documented the problems in the system along with the steps to correct each problem. The report is documented below in Table 2.

Table 2. Documentation of System Problems.

Problem # 1 The system does not have formal incident handling procedures including procedures for: 1. Incident reporting 2. Disaster Recovery and Business Continuity 3. Handling system compromise 4. Handling denial of service (DOS) attacks 5. Handling unauthorized disclosure of sensitive information The impact on the system is that incidents will be reported inconsistently, leading to increased response times to handle the incident. In addition, the company may not be rapidly or adequately informed of incidents of system compromise or unauthorized disclosure of sensitive information, placing the company at greater risk of loss.
Action: To address the incident handling procedures, the Incident Response and the Business Continuity committees will develop and implement the following policies and procedures: 1. A high level Business Continuity planning policy (Appendix B) 2. Procedures for Disaster Recovery and Business Continuity for the production and development systems. 3. Procedures for incident handling and reporting the incident to the Corporate Information Security Unit with specific procedures to handle the following events: a. system compromise b. denial of service (DOS) attacks c. unauthorized disclosure of sensitive information
Steps to Implement the Controls:

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

To establish the incident handling procedures, the teams established a standard six step process[9] (1) Preparation, (2) identification, (3) Containment, (4) Eradication, (5) Recovery and (6) Lessons Learned. For the preparation phase the team completed the following steps:

1. The team established recovery plans for the production, CVS/NIS and logging server (see below)
2. The team pulled together a list of contacts that included the system administrator, a member of the Corporate Information Security team and two of the managers from the web development unit.
3. The team created an escrow envelope of the root passwords and PKI encryption back-up keys (see below)
4. The team created a series of checklists to follow to establish the type of incident in the Identification stage. The checklists handled the three basic risks and in outline can be summarized as:

    a. system compromise – If the system administrator is fairly certain that a system compromise has occurred, then Corporate Information Security should be immediately notified. The system administrator should notify the managers and wait for Information Security before performing actions on the servers. Depending on the nature of the compromise, Information Security may pull the system off of the network for the Containment phase
    b. denial of service (DOS) attacks – If the system administrator has detected a DOS, then Corporate Information Security should be immediately notified. The system administrator should monitor the logs to determine the IP(s) of the attacking system and provide this information to Information Security
    c. unauthorized disclosure of sensitive information – If this event is suspected, the System Administrator will corporate with Corporate Information Security to audit the system logs.

5. Since Information Security had pre-established checklists for handling the containment phase, the team developed checklists for eradication and recovery:
    a. After being given the all clear from Information Security, the team would have a debriefing session with Information Security to determine the root cause.
    b. Based on the root cause analysis, the system would be restored, validated and monitored for further problems.
    c. Finally the team would meet again with Information Security for a follow-up to develop and incident report and improve on the handling process.

6. Next, the team worked with the Awareness and Training Committee to set up an awareness program among the developers to spot incidents and report them to the incident handling team.

[9] SANS Security Essentials, 2003 – Volume One, version 2.1

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

7. Finally, the entire team attended a class in incident handling training from an outside security education program.

Disaster Recovery and Business Continuity:

1. The Business Continuity committee drafted the Business Continuity planning policy (Appendix B)
2. Using the Asset Inventory, the Business Continuity committee determined the level of protection for each asset and wrote procedures for business continuity:
    a. Although the production system is important, the main objective is that the proprietary information is protected. Because of this information on the back-ups are to be encrypted (see below).To allow for full recovery of the servers, one full system back-up per week will be sent to an offsite storage facility that is already contracted with the company. The systems that will be backed up are the Production Web Server, the NIS/CVS server and the Central Logging server.
    b. Daily back-ups of production code and code stored in CVS will be encrypted and piped over SSH for storage on the central logging server.
    c. In the event that the production server is destroyed or non-operational, the team will format the Load Test server with the production back-up and establish a new production server.
    d. The system administrator will perform a full system back-up of production on the Load Test server once per year to verify that the back-ups are being performed correctly.
    e. Development servers will be rebuilt using a set of standard Linux images, but will not be backed up because the important code is stored in CVS.
3. The Business Continuity committee established a training program (through the Awareness and Training Committee) for the system administrator and his back-up so that in the event that one was unavailable, the other could restore the system.

Problem # 2
The system does not have documented operating procedures, including procedures for:

1. System Back-Up
2. Change Control
3. Patching
4. Proper Shutdown
5. Deployment of Production Code

The impact on the system is that the system operations, including back-up, patching, change control, and deployment may be inconsistently executed over time. In addition, this makes it difficult to appoint new system administrators because the system operations are undocumented.

Action:
To address this problem, the Systems Operations committee will draft formal procedures to address proper system management, including system shutdown, back-

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

up, change control, patching and deployment of production code. The Systems Operations committee will work with the Business Continuity team to ensure that the back-up schedule is in line with disaster recovery plans.

Steps to Implement the Controls:

1. Using the procedures drafted from the Business Continuity planning committee, the Systems Operations committee created operational procedures for system back-ups (which are presented in summary):
    a. The system administrator will establish a cron job to tar the full system onto tape once per week for off-site storage. The cron will tar the entire system, encrypt the archive using the server's public key (see below) and then write the archive onto tape. Offsite Back-up tapes will rotated every fourth tape.
    b. The system administrator will establish cron jobs to tar and encrypt the CVS and production web code. The archives will be copied over ssh to the Central Logging Server for storage. The archives will be rotated every seventh archive.
2. Next, the Systems Operations committee created operational procedures for change control (presented in summary):
    a. The system administrator will fill out a change control procedure form documenting the change.
    b. The system administrator will schedule a time for the change and inform the development team of the scheduled time.
    c. The system administrator will back-up the system files that will be affected.
    d. The system administrator will test the change (if possible) in development
    e. The system administrator will then conduct the change.
3. Third, the Systems Operations committee created operational procedures for system patching (presented in summary):
    a. The system administrator will participate in an e-mail alert list for new vulnerabilities.
    b. When a new vulnerability of high or medium severity is announced, the administrator will fill out a change control procedure form and schedule a time for the patch and inform the development team of the patching cycle.
    c. The system administrator will back-up the system files that will be affected.
    d. The system administrator will begin patching in development and test the patch over the course of several days before proceeding to production.
    e. Exceptions to the patching cycle may be approved for emergency cases with the Systems Operations Committee.
    f. To check compliance, a Nessus will be installed on the Central Logging Server and monthly audits of the system will be conducted automatically and the results sent to the Systems Operations Committee.
4. Fourth, the Systems Operations committee created operational procedures for system shutdown (presented in summary):
    a. The administrator will schedule a system down-time and inform the team

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

prior to shutting down the server.

   b. In the event of an emergency shutdown, the system administrator will inform the development team as soon as possible after the shutdown.

5. Last, the Systems Operations committee created operational procedures for deployment of production code(presented in summary):
   a. The project manager will send an e-mail detailing the production time, and deployment instructions.
   b. The systems administrator will back-up the current production code and deploy the new code using the release tag from CVS.
   c. The administrator, project manager and developer(s) will review the deployment for issues. If one is encountered, the system administrator shall restore the production code from back-up.
6. The Systems Operations committee established a training program (through the Awareness and Training Committee) for the system administrator and his back-up so that in the event that one was unavailable, the other could maintain the system.

Problem # 3

The system does not have encryption standards and procedures for back-up media or network controls.

The impact for the system is that encryption may be used inconsistently to secure sensitive information (such as outdated or small key protocols) The result is that system accounts may be compromised using passwords sniffed from network. In addition, if back-up media is stolen, sensitive information may be disclosed.

Action:

To address this problem, the System Operations committee will work with the Corporate Information Security team to draft standards for data encryption.

Steps to Implement the Controls:

1. For back-up media the System Operations committee will use the PKI infrastructure that the company already possesses as the standard. It uses 128 bit keys. The committee drafts standards and procedures based on the existing infrastructure:
   a. Procedurally, the server will use a set of java tools (that are included with the PKI infrastructure) and using the server's public key will encrypt the archive.
   b. The server's private keys will be kept on the certificate authority and also in locked-up escrow disks for disaster recovery.
2. Similarly, the committee has chosen that symmetric encryption keys for SSH and SSL will be at least 128 bit keys. The committee incorporates this into the encryption standards.

Problem # 4

The system does not have standards and procedures for user management, including:

1. formal user registration and de-registration.
2. allocation of system privileges.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

3. handling and use of user credentials, such as passwords.
4. the periodic review of access rights.

Action :
To deal with this issue, the Security Policy and Risk Management Committee will create a high level Access Control policy (Appendix B) and develop underlying standards and procedures to address user management. The Awareness and Training Committee will be responsible for developing user awareness and training plans so that users know how to properly register and so that data owners properly review access rights.

Steps to Implement the Controls:

1. The Security Policy and Risk Management Committee developed and implemented the Access Control policy (Appendix B).
2. The committee creates a standard registration form for access to the servers that would be used for registration of developers, testers, managers and system administrators. The form contains:
    a. A description of user rights and privileges depending on the user's role (developer, tester, manager or system administrator).
    b. A statement that asks the user to keep their password confidential and to change their initial (temporary) password.
    c. The form must be signed by the user's supervisor.
3. The committee creates a standard registration form for access to the proprietary information sites that have access controlled. The form contains:
    a. A description of user rights and privileges.
    b. A statement that asks the user to keep their password confidential.
    c. The form must be signed by the content owner of the web site, and is then passed on to system administrator.
4. The forms are placed on the Production Web Server with links so that users can access and print them off if they need to obtain access to the system.
5. The Committee develops procedures of allocation of system privileges based on the user role:
    a. Developers – will have read access to production code that does not contain proprietary information (unless the developer is working on a site that contains proprietary information). Developers have read/write access to the content on the development environment. They will also be able to read/write code in CVS(on the NIS/CVS server).
    b. Testers – will have the same privileges as developers.
    c. Contractors - will have the same privileges as developers, except that they will not be given access to the production server except for limited access to trouble-shoot code they may be involved in working on.
    d. Managers will have the same privileges as developers.
    e. System Administrators will have root access to the entire environment.
6. Along with description of user rights, the Committee will include instructions for password management. The instructions are derived from company information security standards that are outlined below:
    a. Each user must have their own account. Accounts may not be shared.
    b. Passwords must be a minimum of 8 characters in length and contain both alphanumeric and special characters (including punctuation).

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

c. Passwords must expire every 90 days.
d. Root passwords must be changed every 30 days.
e. Passwords must not be stored in readable files or scripts.

7. Finally, the committee has created standards for periodic review of accounts:
    a. Employee accounts must be reviewed by the content owner or system administrator once be year. Accounts that are no longer needed must be removed.
    b. Contractor accounts must be reviewed by the content owner or system administrator every six months (because contracts usually last 6 months to 1 year). Accounts that are no longer needed must be removed.
8. After the creation of the standards, the Awareness and Training Committee has created a general awareness program to inform users of their access rights and where to register to gain access to the web sites.

Problem # 5

The system does not have written procedures for proper application development including:

1. Code Testing Procedures
2. Checkpoints to confirm information classification and access control
3. Change control procedures and code deployment procedures

One impact on the system is that sensitive information that should have controlled access may go unprotected. A second impact is that if the code is improperly tested or improperly deployed, the application may not work properly and be unavailable to users

Action:

Using input from the web development team, the Systems Operation Committee will create and implement a high level policy on Security Engineering in the Systems Development Life Cycle (Appendix B). In addition, the team will draft development procedures addressing testing, information security checkpoints, development change control procedures and development code deployment procedures.

Steps to Implement the Controls:

1. The Systems Operation Committee develops and implements the policy Security Engineering in the Systems Development Life Cycle (Appendix B).
2. Using documentation and interviews from the development team, the committee develops and publishes formal development procedures that are described in section I. The Current Environment (see figure 3). With the procedures, developers are required to:
    a. Test the code on the User Acceptance Testing (UAT) server and have this documented prior to proceeding to a production release.
    b. Store all production code in CVS and to tag code prior to release to UAT, Load Test and Production.
3. With the procedures, managers are required to
    a. confirm content sensitivity as the web site requirements are gathered.
    b. document that security requirements have been met in the UAT stage, prior to production.
4. With the procedures, the system administrators is required to:

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

a. Confirm that documentation of the security requirements is in place prior to deploying content. b. After deployment alerting the developer and project manager to review the web site to make certain that the code is functioning properly. 5. Working with the Awareness and Training Committee, training documentation for new system administrators, developers and managers will be prepared to inform them of the procedures.

Problem # 6 The servers in the ISMS are not regularly patched. The impact is that the servers can be easily compromised by internal hackers or worms.
Action: To address this issue, the System Operations committee will draft patching procedures (above). The system administrator will be responsible for downloading rpms (red hat package manager) and installing them onto the servers.
Steps to Implement the Controls: 1. The system administrator will implement a patching program based upon the procedures above. 2. The system administrator will install nessus[10] upon the Central Logging Server (see below) and automate the scans. Results of the scans will be mailed to the Systems Operations committee.

Problem # 7 The system does not use controls to safeguard the integrity and confidentiality of data passing over the network. The impact is that confidential information including usernames and passwords may be sniffed off of the network and used to compromise accounts.
Action: To address this issue, the Systems Operations Committee will select network controls, based on the encryption standards, to secure the information. The system administrator will implement the controls in the working systems.
Steps to Implement the Controls: 1. The system administrator will audit the servers using nmap and determine current system services. 2. The system administrator will shut down and disable unneeded services on the servers. 3. Using the encryption standards, the system administrator will install secure versions of necessary services: a. The NIS server will be converted to a Kerberos Key Distribution Center (KDC) and manage central authentication for the environment. b. Telnet will be replaced by Secure Shell (SSH)

[10] Nessus is an excellent (free) vulnerability scanner available at www.nessus.org

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

c. File Transfer Protocol (FTP) will be replaced by Secure FTP (over SSH).
d. For web sites requiring authentication, secure socket layers (ssl) will be used to protect user names and passwords.
e. CVS will be accessed over SSH to protect the content.

Problem # 8

The system does not use controls to safeguard the integrity and confidentiality of data stored on removable media, such as back-up tapes.

The impact on the system is that if an attacker can obtain the removable media, then they will potentially have access to sensitive information from the production server.

Action:

To address this issue, the Systems Operations Committee will select encryption controls for the removable media based on the encryption standards. The system administrator will implement the controls in the working systems.

Steps to Implement the Controls:

1. The Systems Operations Committee has chosen to utilize the existing company PKI infrastructure. They register each server into the certificate authority.
2. For each server, they obtain a public key. The systems administrator installs the public key onto the server along with a java-based tool that was included with the PKI infrastructure.
3. The systems administrator creates cron jobs to automate the back-up process.
    a. For the Production Web Server, CVS/NIS server and the Central Logging Server (see below). Offsite back-ups are created once per week. The full system is backed up using tar. The tar is encrypted using the public key and then written onto the back-up tape.
    b. The tape handlers are instructed to use 8 tapes in the rotation (so that 8 weeks are preserved).
    c. For the Production Web Server the production web content is backed up using tar and transferred (over SSH) to the Central Logging Server every 24 hours.
    d. For the NIS/CVS Server the development code in CVS is backed up using tar and transferred (over SSH) to the Central Logging Server every 24 hours.
    e. On the Central Logging Server, the transferred back-ups are rotated so that only 1 week of back-ups is preserved.
4. The systems administrator makes a schedule to confirm that back-ups are functioning correctly ever six months (see audit checklist)
5. The systems administrator makes an escrow copy of the private keys (including the java tools to decrypt) and makes a checklist for recovering the data. This is included in the disaster recovery planning.

Problem # 9

The system does not utilize a tool to monitor:

1. system logs for suspicious activity, such as an Intrusion Detection System (IDS)
2. the activities of authorized user for audit purposes

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

3. system performance for DOS attacks or application failure

The impact on the system is that suspicious activity is less likely to be detected and response will be slower. In addition, because activities of authorized users are not being monitored, it is difficult to establish suspicious activity on their accounts, such as when the account has been compromised.

Action:
The Systems Operations Committee will select log watching and monitoring tools for the system. In addition, the committee will select a smaller server ($ 10000) to act as a central log repository system. The system administrator will implement the controls and the Audit and Compliance Committee will be responsible for scheduling audits of the syslogs

Steps to Implement the Controls:

1. The Systems Operations Committee purchases a small server to act as a Central Log Repository. The server is configured to accept the system logs from the from the other machines using the syslog daemon. In addition, Nessus is installed on the Central Log Repository and automated to provide audit scans as indicated in part Z.
2. Swatch (Simple Watcher) is selected as part of the monitoring system selected by the Systems Operations Committee to monitor the system logs and send alerts.
3. The systems administrator installs Swatch and using past incident logs, configures Swatch to send an e-mail to him and his back-up based upon certain log alerts. Swatch is installed on all of the machines in the environment (production and development).
4. The Systems Operations Committee also has selected Snort to monitor the system for suspicious activity.
5. The systems administrator installs Snort and tunes the configuration by using nessus to run common attacks on the servers. Snort is installed on all of the machines in the environment (production and development).The system administrator configures Snort to log suspicious activity to the syslog.
6. Next, so that the machines maintain proper time and date to track incidents, the systems administrator installs the network time protocol (ntp) service on all machines and synchronizes them to a time server that the company maintains.
7. The systems administrator links the e-mail alerts to a pager so he can receive the alerts 24 x 7. The pager is rotated between the system administrator and his back-up.

Problem # 10
The system does not utilize a tool to monitor the integrity of system files and web content files.

The impact on the system is that the system administrator will not be aware of malicious or accidental changes to important files on the system.

Action:

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

To address this problem, the Systems Operations Committee will select integrity checking software and the systems administrator will implement the software on the system.
Steps to Implement the Controls: 1. The Systems Operations Committee has selected Tripwire and prepares operating procedures for the systems operations documentation: a. The system binaries and configuration files (/etc) will be hashed for the production, CVS and Central Logging Servers and burned onto CD. b. The hashes will be audited once every six months or in prior to system configuration change. After the change, a new signatures CD will be created. c. Tripwire will also run on the system and send alerts to the syslog regarding changes in system binaries, configuration files and web content. d. The system administrator will investigate and report any alerts generated that are not part of a normal system configuration change or deployment. 2. The systems administrator installs tripwire onto the Production (web), CVS and Central Logging Servers and creates the initial signatures CDs.

Problem # 11 Anti-virus software is not installed on the servers of the system The impact to the system is that viruses and worms may not be detected, leading to system compromise and potentially destruction of important data.
Action To tackle this problem, the Systems Operations Committee will select anti-virus software and the systems administrator will implement the software on the system.
Steps to Implement the Controls 1. The Systems Operations Committee selected a commercial Anti-Virus scanner for Linux. 2. The systems administrator installed the scanner on all of the machines in the environment (production and development). 3. The systems administrator configured cron jobs to update the virus signature files each night and scan the machines. 4. The summary results of the scans were logged in the syslog and the administrator was sent an alert if a virus was detected.

Statements Of Applicability

For the system, the committee also drafted statements of applicability (three of which are documented below):

1. Electronic Commerce Security (ISO 8.7.3)

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

The audit checklist states "Whether Electronic commerce is well protected and controls implemented to protect against fraudulent activity, contract dispute and disclosure or modification of information."[11]

Statement
The Information Security Management System is an internal company system that is not used for electronic commerce. Although certain security controls (such as user authentication and authorization) will still apply, this section in not applicable, because the ISMS does not handle electronic commerce.

2. Event Logging (ISO 9.7.1)

The audit checklist states "Whether audit logs recording exceptions and other security relevant event are produced and kept for an agreed period to assist in future investigations and access control monitoring."[12]

Statement
To monitor user access and to assist in investigations of system abuse, the system maintains event logs. All of the event logs are centralized to a logging server and stored onto tape back-up for a period of two months. The logs are scanned by automated tools to alert the system administrator of suspicious activity.

3. Separation of development and operational facilities (ISO 8.1.5)

The audit checklist states " Whether the development and testing facilities are isolated from operational facilities. For example, development software should run on a different computer to that of the computer with production software. Where necessary development and production network should be separated from each other." [13]

Statement
The primary function of the ISMS is to support development of web applications. To support this function and comply with the standard, the ISMS has segregated the development and production environments. The development environment is physically separated (a locked computer lab) from the production environment (data center) and development code must undergo rigorous testing prior to being introduced into production.

IV. Check – System Auditing

[11] Information Security Management BS 7799.2:2002 Audit Check List for SANS (2003) Val Thiagarajan.
[12] Information Security Management BS 7799.2:2002 Audit Check List for SANS (2003) Val Thiagarajan.
[13] Information Security Management BS 7799.2:2002 Audit Check List for SANS (2003) Val Thiagarajan.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Working through the Audit and Compliance committee, the ISMS steering committee created a checklist to audit the system against specific criteria that were identified as risks to the system (see risk analysis). The checklist was developed against the controls that were implemented to mitigate the risks on the system. The checklist was distributed to all of the committees and the systems administrator.

As technical controls are implemented on the system, the administrator will use the checklists to verify that the controls are operating properly. In addition, the checklist will be incorporated into systems procedures and systems validation documentation. This will allow the systems administrator to train back-up personnel in proper system procedures and will also train the team on what to expect in a yearly audit. An outline of the checklist for all of the major controls from the risk analysis is included in Appendix F – Extended Audit Checklist.

Because of the proprietary information stored on the production server, one of the primary areas of concern was User Access Management (ISO 17799 section 9.2) and the detailed audit checklist for this section is presented in Table 3 – Audit Checklist for User Access Management. This detailed checklist was produced from the outline (Appendix F) and will be used to ensure that access to proprietary information is controlled.

The Audit Checklist for User Access Management (Table 3.) is an improvement for the system and will be used to ensure that (1) users are being properly registered and de-registered; (2) that privileges are assigned on a need-to-use basis; (3) that user passwords are kept confidential and handled properly; and (4) access rights will be periodically reviewed to remove unneeded accounts. These procedures will reduce the risk of proprietary information loss. The system will be audited on a yearly basis and the results will be reported to the Information Security Management Steering Committee (see section V).

Table 3. Audit Checklist for User Access Management[14]

User Access Management (ISO 17799: 9.2)
Objective: To reduce the risk of proprietary information loss, the system will have procedures for 1. formal user registration and de-registration. 2. allocation and management of system privileges. 3. handling and use of user credentials, such as passwords. 4. the periodic review of access rights. A user awareness program will be established so that users are aware of their rights and responsibilities. The system will be periodically audited to ensure that procedures are being implemented upon the systems.

[14] This section uses information from the Information Security Management BS 7799.2:2002 Audit Check List for SANS (2003) Val Thiagarajan.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

1. User registration and de-registration		
Objective: Establish whether the system has a formal system of user registration and de-registration. The registration system regulates who has access to the system and reduces the risk of proprietary information loss.		
Item	Question	Audit Steps
1.1	Does the system have formal user registration and de-registration procedures?	1. Identify the formal user registration and de-registration procedures. 2. Verify that there is an owner of the procedures to review and update them.
1.2	Does each web site that contains sensitive information have formal user registration and de-registration procedures?	1. Review the web sites on the system and determine which ones are classified as restricted. 2. Identify the formal user registration and de-registration procedures for those sites. 3. Verify that there is an owner of the procedures to review and update them.
1.3	Are the procedures current?	1. Check the last review/modification date of the procedures. 2. Verify that the procedures have been reviewed in the past year.
1.4	Are users aware of the registration procedures?	1. Interview active end-users and verify that they have registered using the system forms.
1.5	Has each active user filled out the application form?	1. Examine the /etc/passwd file (less /etc/passwd) and compile a list of active users. 2. For web sites, compile the listing from the .htaccess file. 3. Verify that the list of active users have filled out the form.
1.6	Do all authorized users have unique identifiers (such as user accounts)?	1. Sort the list(s) compiled in 1.5, and verify that all user accounts are unique entries.
1.7	Do any contractors have access to the production systems?	1. Using the list(s) of users from 1.5, sort through the company directory and verify that none of the active user are contractors.
1.8	Has access been removed for	1. Using the list(s) of users from

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

	users who have left the company?	1.5, sort through the company directory. 2. Confirm that the current list of users are active employees.

2. Allocation and Management of Privileges		
Objective: Establish that the system has procedures for the allocation and management of system privileges. Privileges must be allocated on a need-to-use basis and after a formal authorization process. The management of privileges limits abuse of the system by authorized users.		
Item	Question	Audit Steps
2.1	Does the system have formal procedures for the allocation and management of system privileges?	1. Identify the allocation and management procedures. 2. Verify that there is an owner of the procedures to review and update them.
2.2	Are the procedures current?	1. Check the last review/modification date of the procedures. 2. Verify that the procedures have been reviewed in the past year.
2.3	Are users informed of their privileges?	1. Examine the user registration form and make certain that it contains the description of user rights and privileges. 2. Interview an end-user and verify that he/she is aware of the rights and privileges.
2.4	Does the system contain accounts that are shared? (This makes auditing of user access impossible)	1. Examine /etc/passwd (less /etc/passwd) and verify that each registered user has only a single account.
2.5	Do any developers, managers or testers have administrative rights (such as root access)?	1. Examine /etc/passwd (less /etc/passwd) and verify that there are no developers, managers or testers have an UID 0 a. (for example mark:x:0:0:Mark Uno:/home/mark:/bin/bash)[15] 2. Examine /etc/group (less /etc/group) and verify that there are no developers, managers or

[15] Drawn from SANS Security Essentials Volume Two, version 2.1.(2003)

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

		testers in an administrative group.
2.6	On production, are developers, managers and testers restricted from write access to production web content?	1. Change directory (cd) to the production content folders. 2. Using ls –l, verify that the developers, managers and testers have only read access to non-proprietary information a. Read access looks like: drwxr-x--- 2 root devteam, where devteam is the group that contains the developers, managers and testers.
2.7	On production, do only system administrators have read access to proprietary information on the web server?	1. Change directory (cd) to the production content folders. 2. Using ls –l verify that only root has access to sites containing proprietary information (unless a developer is working on the site). a. Access privileges look like: drwxr-x--- 2 root root.

3. Handling and use of user credentials, such as passwords		
Objective: Establish whether the system has formal management processes to govern password management, so that user accounts have a reduced risk of being compromised. This reduces the risk of proprietary information loss.		
Item	Question	Audit Steps
3.1	Does the system have formal procedures for the handling of user credentials?	1. Identify that the procedures exist. 2. Verify that there is an owner of the procedures to review and update them.
3.2	Are the procedures current?	1. Check the last review/modification date of the procedures. 2. Verify that the procedures have been reviewed in the past year.
3.3	Do users sign a statement to not share their password and to keep the password confidential?	1. Obtain the user registration form for the system. 2. Verify that the statement is present in the user responsibilities.
3.4	Are users required to change their initial temporary password?	1. Obtain written permission from senior management to audit temporary passwords. 2. Obtain a list of the temporary user

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

		passwords from the systems administrator. 3. Test the temporary passwords against the user accounts.
3.5	Do all accounts possess passwords?	1. Examine /etc/shadow (less /etc/shadow) 2. Verify that all user accounts have passwords.
3.6	Must administrative account passwords (such as root) be changed every 30 days?	1. In /etc/shadow, verify that root passwords must expire in 30 days a. The root password accounts should look like: <username>:<password>:<last changed>:<may change>:30:<warn>:<disable>:<expire date>:[16]
3.7	Must regular user passwords be changed every 90 days?	1. In /etc/shadow, verify that other passwords must expire in 90 days a. The user accounts should look like: <username>:<password>:<last changed>:<may change>:90:<warn>:<disable>:<expire date>:[17]
3.8	Have any users chosen dictionary passwords?	1. Obtain written permission to run John the Ripper on the password files from senior management. 2. Combine /etc/passwd and /etc/shadow using the unshadow tool. 3. Run John the Ripper on an isolated machine for 24 hours. 4. Report the findings to senior management 5. (Ask the users to change dictionary based passwords).

4. Periodic review of access rights		
Objective: Establish whether there are processes to review the access rights of accounts. This should be done on a periodic basis to remove unneeded accounts and reduce risk of proprietary information loss.		
Item	Question	Audit Steps

[16] Drawn from SANS Security Essentials Volume Two, version 2.1.(2003)
[17] Drawn from SANS Security Essentials Volume Two, version 2.1.(2003)

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

4.1	Does the system have formal procedures for review of access rights?	1. Identify that the procedures exist. 2. Verify that there is an owner of the procedures to review and update them.
4.2	Are the procedures current?	1. Check the last review/modification date of the procedures. 2. Verify that the procedures have been reviewed in the past year.
4.3	Have the content owners of reviewed the access control lists in the past six months?	1. Contact the content owner of each site and verify that the access control list has been reviewed in the past six months. 2. On the production server change directory (cd) to the directory with the .htaccess file. 3. Use ls –al to determine the last modification date (is it in the last six months?) 4. Open the file (less .htaccess) and using the company directory determine if the listed persons are active employees.
4.4	Has the password file on the production web server been reviewed in the past year, or as members have left the team?	1. Contact the system administrator and verify that the file has been reviewed in the past year (no contractors are allowed access to production) 2. On the production server change directory (cd) to /etc 3. Use ls –al passwd to determine the last modification date (is it in the less than one year?) 4. Open the file (less passwd) and using the company directory determine if the listed persons are active employees.
4.5	Has the password file for the development environment been reviewed in the past six months, or when members have left the team?	1. Contact the system administrator and verify that the file has been reviewed in the past six months. 2. On the production server change directory (cd) to /etc 3. Use ls –al passwd to determine the last modification date (has it been modified in the last six months?)

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

		4. Open the file (less passwd) and using the company directory determine if the listed persons are active employees. 5. Make a list of the current development team and verify that only active members have accounts.

V. Continuous Improvement ("Act" Phase)

Improving the System Through Lessons Learned from Incident Handling

The incident handling procedures follow the traditional stages of (1) Preparation, (2) Identification, (3) Containment, (4) Eradication, (5) Recovery and (6) Lessons Learned. During the lessons learned phase, the Incident Handling team will use the fault trees developed in section II (Appendix C) and determine if the incident is a fault in the original fault tree. If the fault exists, then the team will use the flash cards to examine how the fault occurred by expanding the event tree to determine where the controls failed. If the fault does not exist in the tree (or the flash card does not exist), then the team will construct a new flash card, including a new event tree to determine where the controls failed.

Every month, the Incident Handling team will send a report to the Security Policy and Risk Management Committee detailing the incidents and the incident severity as compared to past incidents in the plans for risk management (section II). The committee will review the report and make a judgment on events that should be controlled. The Security Policy and Risk Management Committee will then construct a problem statement and action plan. If the control is considered a major capitol purchase, such as buying a new server, or if the change involves a major shift in policy or procedure, ther action plan will be reviewed and approved by the ISMS committee,

The Security Policy and Risk Management Committee will work with the other committees to implement the control. Finally, the Audit and Compliance committee will construct another entry into the audit checklist so that the effectiveness of the control can be evaluated in audits.

Improving the System through Auditing

Working through the Audit and Compliance committee, the ISMS committee will schedule yearly internal review cycles for the system. First, the asset inventory will verified and updated. Following the update, the system procedures will be reviewed and updated. In addition, to the system procedures, the owners of the high-level policies will

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

review and recommend updates to the policies that will be reviewed by the Research Information Management Committee. Fourth, the Audit and Compliance committee will review the system using the checklist developed in section IV (Appendix F) and prepare a detailed statement of findings. Fifth, the Security Policy and Risk Management Committee will perform a gap analysis using the ISO 17799 general checklist to determine if the system has other gaps that need consideration. The report from the Audit and Compliance committee and the gap analysis will be presented to the Information Security Management Steering Committee with recommendations on improvements for the system. The ISMS steering Committee will give a decision on the recommendations and the system improvements will be delegated to the appropriate committees for implementation.

In addition to the yearly audit cycle, the Information Security Management Steering Committee will schedule an outside audit every three years with and independent audit company. The outside company audits the system against ISO 17799 and the checklist developed in section IV. The audit results will be presented to the Information Security Management Steering Committee who will prioritize the findings and communicate them to the committees so that the committees can recommend controls. The committees will prepare an action plan that will be presented to the ISMS steering committee and based on the decisions, will implement controls to mitigate the risks highlighted in the independent audit.

End Note

The reviewers noted that using facts contained within this paper that the actual company might be discerned. So that the company would clear this manuscript, I have written on an environment that has been retired (5 months prior to me starting this manuscript). This paper assumes that the environment is still active and is presented in the state it was prior to its retirement.

Bibliography

1. "Authentication, Authorization, and Access Control". Apache Documentation (version 1.3), Apache Foundation. URL: http://httpd.apache.org/docs/howto/auth.html. 2003.

2. Cederqvist 1.11.17 Manual for CVS. Concurrent Versions System. URL: https://www.cvshome.org/docs/manual/. 2004.

3. Thiagarajan, Val. "Information Security Management BS 7799.2:2002 Audit Check List for SANS". In Track 11 – SANS 17799 Security and Audit Framework 11.5. SANS Institute. 2003.

4. Wood, Charles Cresson. Information Security Policy Made Easy – 8th Edition, , PentaSafe, Sausalito, California. 2001

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

5. Information Technology – Code of practice for information security management (BS ISO/IEC 17799:2000 BS 7799-1:2000) British Standards Publishing Limited (BSPL). 2000.

6. Pharmaceutical Industry Profile 2004. The Pharmaceutical Research and Manufactures of America (PhRMA) URL: http://www.phrma.org/publications/publications//2004-03-31.937.pdf. 2004.

7. Cole, Eric, Fossen, Jason, Northcutt, Stephen, Pomeranz, Hal. Sans Security Essentials with CISSP CBK (Volume 1 – Version 2.1),. The SANS Institute. 2003.

8. Cole, Eric, Fossen, Jason, Northcutt, Stephen, Pomeranz, Hal. Sans Security Essentials with CISSP CBK (Volume 2 – Version 2.1),. The SANS Institute. 2003.

9. "Trends in Proprietary Information Loss (September 2002) – Survey Report" Sponsored by Price Waterhouse Coopers, U.S. Chamber of Commerce and ASIS Foundation. 2002.

10. "Why Do Prescription Drugs Cost So Much? and Other Questions About Your Medicines" The Pharmaceutical Research and Manufactures of America (PhRMA) URL: http://www.phrma.org/publications/publications/brochure/questions/whycostmuch.cfm. 2004.

Software

1. Concurrent Versions System (CVS) is used to manage change control during software development. It can be found at https://www.cvshome.org/

2. Nessus is freely available vulnerability scanning software and can be found at www.nessus.org

3. Tomcat is a java-based web server and can be found at http://jakarta.apache.org/tomcat/index.html

Appendix A – Extended Asset Classification

Production Apache/Tomcat Web Server and Database Server
The production server is a quad-processor server that hosts a web server (Apache) with java support (Tomcat) and a database server (MySQL). The production server hosts approximately 40 web sites. Two of the web sites contain proprietary information that is moved by departmental administrators to the server using FTP. In addition, the MySQL database supports a patch tracking software. Upon boot, all desktops in the environment transmit information to an

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

application on the server, which stores it in the database.
Immediate value: 1. Server Hardware: $ 50,000 2. Software (Installation time) $ 1000
Potential Impacts to Confidentiality: Disclosure of information contained in Web Sites: - Proprietary Research 1. Disclosure of Proprietary information - Desktop Deployment Information 1. Disclosure of Patch level of environment - Divisional IT Strategy 1. Disclosure of organizational information - Divisional Policy, Training and Procedures 1. Disclosure of organizational information - Departmental Specific Information 1. Disclosure of organizational information Potential Impacts to Integrity: 1. Corruption of database (regenerated daily) 2. Corruption of production source code (integrity) Potential Impacts to Availability: 1. Ability for internal business units to use server is impaired (availability)
Likely consequences: Disclosure of information contained in the web sites: 1. Disclosure of Proprietary information (confidentiality) - Loss of competitive advantage (potentially hundreds of thousands of dollars to hundreds of millions of dollars) 2. Disclosure of Patch level of environment (confidentiality) - Attacker gains detailed vulnerability levels of the environment 3. Disclosure of organizational information (confidentiality) - Attacker gains information useful for social engineering Impacts to Integrity: 1. Corruption of database 2. Deployment teams cannot track patch progress. 3. Data is regenerated daily 4. Corruption of production source code 5. Ability for internal business units to use server impaired 6. Server may be more easily compromised Impacts to Availability 1. Ability for internal business units to use server impaired (availability)

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

2. Loss of internal user confidence 3. Productivity losses 100 employees per hour x 5% time (5 people/hour)

Production CVS/NIS Server
The CVS/NIS server is a dual-processor server that hosts the configuration management system concurrent versions system (CVS). The server also hosts an anonymous FTP server that allows the developers access to team tools. Finally, the server hosts a NIS server that provides user authentication for all of the other systems except for the Production Apache/Tomcat Web Server and Database Server. By procedure, the proprietary information stored on the production server is not stored in CVS.
Immediate value: 1. Server Hardware: $ 50,000 2. Software (Installation time): $ 1000
Potential Impacts to Confidentiality: 1. Compromise of the NIS server accounts 2. Disclosure of information and source code contained in Web Sites: a. Proprietary Research: i. No impact: By procedure, this information is not stored on the CVS server b. Desktop Deployment Information: i. No impact: This information is contained in the database on the production server. c. Divisional IT Strategy: i. Disclosure of organizational information d. Divisional Policy, Training and Procedures: i. Disclosure of organizational information e. Departmental Specific Information: i. Disclosure of organizational information Potential Impacts to Integrity: 1. Corruption of source code used to develop production web sites Potential Impacts to Availability: 1. Disruption of the NIS server 2. Disruption of the CVS server 3. Loss of development source code
Likely consequences: Impacts to Confidentiality: 1. Compromise of the NIS server accounts would allow attacker access to the development and load test server. The NIS server does not authenticate the production web server accounts. 2. Disclosure of information contained in Web Sites: a. Disclosure of organizational information (confidentiality)

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

i. Attacker could gain useful information for social engineering

Impacts to Integrity:

1. Corruption of source code used to develop production web sites:
    a. Trojan code could be used to obtain access to the production server.
2. Corrupted code would have to be recreated by the developers resulting in lost development time.

Impacts to Availability:

1. Disruption of the NIS server would prevent developers from logging into development environment.
2. Disruption of the CVS server would prevent developers from using CVS for source code management.
3. Loss of source code on server would result in lost development time. Ten developers use the CVS server, and this code would be lost up until the previous back-up.

Load Test Apache/Tomcat Web Server
The Load Test Servers is a dual-processor server that hosts a web server (Apache) with java support (Tomcat) and is primarily used to stress-test larger applications. Unlike Production, the Load Testing Server does not host a MySQL database.
Immediate value: 1. Server Hardware: $ 50,000 2. Software (Installation time) $ 1000
Potential Impacts to Confidentiality: 1. Disclosure of Application Source Code used in testing Potential Impacts to Integrity: 1. Corruption of application source code used in testing Potential Impacts to Availability: 1. Disruption of Load Test Activities
Likely consequences: 1. Disclosure of Application Source Code used in testing may disclose a programming flaw that could be used to compromise the production server. By procedure, the Load Test Server only hosts the application being tested, and no other applications (such as the other production applications) 2. Corruption of application source code used in testing – Source code is pulled from CVS, but the changes in test code are made in the development environment. Corruption of the application source code could lead to a disruption of test activity. 3. Disruption of Load Test Activities – The load testing schedule is fairly light for the group and disruptions would last at most an hour of time, unless

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

the server were destroyed. Without the server, the team would use part of development to do some load testing, until the server was repaired.

Tape – Back-up Media:
Immediate Value: $ 10 per tape
Production Apache/Tomcat Web Server and Database Server: 1. Potential Impacts to Confidentiality: a. The back-up contains the same content that is on the production server with similar impacts. 2. Potential Impacts to Integrity: a. Corruption of the information in the back-ups could result in corrupted or unusable code being introduced if the code needs to be reinstalled. 3. Potential Impacts to Availability: a. Loss of back-up media would result in not being able to reestablish the production environment in the event of data-loss.
Production CVS/NIS Server: 1. Potential Impacts to Confidentiality: a. The back-up contains the same content that is on the CVS/NIS server with similar impacts. 2. Potential Impacts to Integrity: a. Corruption of the information in the back-ups could result in corrupted or unusable code being introduced with a reinstall 3. Potential Impacts to Availability: a. Loss of back-up media would result in not being able to reestablish the CVS/NIS environment in the event of data-loss.
Load Test Apache/Tomcat Web Server: 1. Because the information in this environment is drawn from CVS, only the operating system is backed-up.

Development Servers
The development systems contained within the computer lab are Linux-based, and are desktop personal computers with increased memory and larger hard drives than the typical desktop. They run Apache and Tomcat Web Servers. One of them also has a MySQL database.
Immediate value: - Hardware: $ 3,000 - Software (Installation time) $ 200
Potential Impacts to Confidentiality: Disclosure of information and source code contained in Web Sites: 1. Proprietary Research: a. No impact: By procedure, this information is not stored on the development servers 2. Desktop Deployment Information: a. No impact: This information is not stored on development servers

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

3. Divisional IT Strategy:
    a. Disclosure of organizational information
4. Divisional Policy, Training and Procedures:
    a. Disclosure of organizational information
5. Departmental Specific Information:
    a. Disclosure of organizational information

Potential Impacts to Integrity:
1. Corruption of development source code

Potential Impacts to Availability:
1. Development may be disrupted

Likely consequences:

Disclosure of information contained in Web Sites (confidentiality):
1. Disclosure of organizational information
2. Attacker would gain useful information for social engineering

Impacts to Integrity:
1. Corruption of source code used to develop production web sites
2. Trojan code could be used to obtain access to the production server if the attacker could insert the code into the developer's work (this is difficult)
3. Corrupted code would have to be recreated by the developers resulting in the developer having to pull the previous code from CVS and losing the changes (slight lost development time).

Impacts to Availability:
1. Loss of development source code on the servers could result in lost development time. This would be minimal because the developer could pull the source code from CVS and use another development machine.

Network and Switches (and the data that travels across them)
The research division relies heavily on the network switches and ethernet cables that connect the offices to the data center.
Immediate Value: 1. $500,000 for the entire site (several million for the entire company)
Potential Impacts to Confidentiality: Disclosure of user names and passwords to accounts (sniffed off the network). Disclosure of information contained in Web Sites (sniffed off the wire): 1. Proprietary Research: a. Disclosure of Proprietary information 2. Desktop Deployment Information: b. Disclosure of Patch level of environment 3. Divisional IT Strategy: a. Disclosure of organizational information

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

4. Divisional Policy, Training and Procedures:
    a. Disclosure of organizational information
5. Departmental Specific Information:
    a. Disclosure of organizational information

Potential Impacts to Integrity:
1. Corruption of data being transmitted to the deployment database
2. Corruption of development source code being sent to CVS

Potential Impacts to Availability:
1. Ability for internal business units to use the network

Likely consequences:

Disclosure of user names and passwords to accounts (sniffed off the network).
1. This would allow the attacker to access restricted resources and could lead to the disclosure of the proprietary information of the web sites below.

Disclosure of information contained in Web Sites:
1. Disclosure of Proprietary information (confidentiality):
    a. Loss of competitive advantage (potentially $ 100,000 of dollars)
2. Disclosure of Patch level of environment (confidentiality):
    b. Attacker gains detailed vulnerability levels of the environment
3. Disclosure of organizational information (confidentiality):
    c. Attacker gains information useful for social engineering

Impacts to Integrity :
1. Corruption of database data - Deployment teams cannot track patch progress, but this has little effect because the data regenerated daily
2. Corruption of source code going into CVS - This has little impact, because the developer would probably resubmit or correct the code.

Impacts to Availability:
1. Ability for internal business units to use network impaired
    a. Loss of internal user confidence
    b. Productivity losses 1000 employees per hour x 20% time (200 people/hour)

Facility – Data Center
The data center houses several hundred servers, including the servers of the web development team. It is staffed by continuously (24 hours, 7 days per week) by members of networking and the tape back-up operators. The data center occupies its own building and does not share access with other units. Access to the data center is controlled by swipe card access that must be specially authorized by security. Other visitors must sign in and must be escorted at all times. The facility has redundant wiring, a halon fire suppression system and back-up generators.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Immediate Value: 1. $ 20 – 25 million dollars in servers and hardware
Potential Impacts to Confidentiality: 1. Attackers who gain physical access to the servers may gain access to confidential data Potential Impacts to Integrity: 1. Attackers who gain physical access to the servers may corrupt production data. Potential Impacts to Availability: 1. Attackers who gain physical access to the servers may disrupt normal business operations by shutting down servers, or the network.

Facility – Development Lab
The development lab hosts four development systems that are stored in a locked room. The system administrator and each of the managers have a key to the lab.
Immediate Value: 1. $ 20,000 dollars in hardware
Potential Impacts to Confidentiality: 1. Attackers who gain physical access to the servers may gain access to development source code. Potential Impacts to Integrity: 1. Attackers who gain physical access to the servers may corrupt development source code Potential Impacts to Availability: 1. Attackers who gain physical access to the servers may disrupt development operations

Personnel The company has an intense personnel screening process that includes checks of the employees professional and academic credentials, drug screening, and identity and background checks. Employees and contractors are also required to sign a nondisclosure agreement regarding the work they perform at the company.
Managers: The web development team has 3 managers who oversee the development teams. The managers work with the business to collect requirements and manage the projects. Potential risks to confidentiality: 1. The managers have root access to all resources on all servers in the environment.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Potential risks to integrity: 1. With root access, the managers can modify any file on any servers in the environment. Potential risks to availability: 1. When one manager is on leave another manager will fill in for the project, resulting in little impact.
Developers: The web development team has 6 developers who typically work in teams of one or two developers. Potential risks to confidentiality: 1. The developers have access to all resources on all servers in the environment. Potential risks to integrity: 1. The developers have root access on the development servers and can modify any file in development. Potential risks to availability: 1. Development is typically done in teams of two developers, so that when one is one leave, the other one can fill in.
Tester: The unit has a single tester who performs all load and integration testing. Potential risks to confidentiality: 1. The tester has access to all resources on all servers in the environment. Potential risks to integrity: 1. The tester has root access on the development servers and can modify any file in development. Potential risks to availability: 1. When the tester is on leave, testing falls upon the developers, which may slow down during these periods.
Contractors: Typically, the development unit has a couple of contractors who fill in development work during heavy development periods. Potential risks to confidentiality: 1. The contractors have access to all resources on all servers in the environment. Potential risks to integrity 1. The contractors have root access on the development servers and can modify any file in development.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Potential risks to availability 1. When a contractor leaves the company, a developer picks up their project.
System Administrator: The unit has a single systems administrator. Potential risks to confidentiality: 1. The systems administrator has root access to all resources on all servers in the environment. Potential risks to integrity: 1. With root access, the administrator can modify any file on any servers in the environment. Potential risks to availability: 1. When the systems administrator is leave, one of the managers fills in for the deployments, which slow down during these periods.

Appendix B – Policies[18]

Policy – System and Application Access Control (section 9.1 of the ISO 17799 standard)

Purpose:
To protect systems and applications from unauthorized use and to protect the confidentiality of sensitive information.

Scope:
This policy covers all company information systems and applications.

Policy:
All company information systems and applications must have a defined set of procedures that conform to the access control standards as defined by the Corporate Information Security department. At minimum, the access control standards will contain:

1. standards for formal user registration and de-registration.
2. standards for allocation of system privileges.
3. standards for handling and use of user credentials, such as passwords.
4. standards for the periodic review of access rights.

Roles:

[18] This Appendix uses Security Policies Made Easy (2001) and SANS Security Essentials Volume One, version 2.1.(2003)- chapter 8 as references.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

1. Developers are responsible for adhering to the standards during system development.
2. System Administrators are responsible for adhering to the standards for the systems under their control.
3. Data Owners are responsible adhering to the standards and for determining that access control procedures are commiserate with business risk.
4. Corporate Audit is responsible for auditing systems to ensure that they comply with corporate standards.
5. Corporate Information Security is responsible for enforcing the policy. It is also responsible for developing access control standards that reflect industry best practice and are appropriate for the business.

Enforcement:
Systems or applications that are found to be out of compliance with the access control standards may be taken offline by Corporate Information Security until compliance is achieved.

Definitions:
1. System – an organizational structure that is used to manage data or information, such as, but not limited to, a computer information system.
2. Application – A computer program or interface that allows the user to access company data or information.

Policy – Business Continuity Planning (section 11.1 of the ISO 17799 standard)

Purpose:
The purpose of this policy is to provide for planning that assures the continuation of business operations in the event of a disruption or a disaster.

Scope:
This policy applies to systems and assets that are considered essential for business operations.

Policy:
Systems and assets that are essential to business operations must have plans in place to ensure that operations may be resumed in the event of a disaster or disruption. Essential systems will be identified in the business and assessed on a yearly basis. Business data owners in conjunction with system administrators will establish business continuity procedures so that business operations may be resumed after a disaster or disruption. The business continuity procedures will include disaster recovery procedures that are in accordance with recovery objectives designated by the business data owner.

Roles:

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

1. System administrators are responsible for system-level planning on the systems they maintain, including hardware, operating system and shared applications, such as web servers and databases that are shared between system users.
2. Business Data owners are responsible for proper classification of the data as well as coordinating with the system administrators to insure system level controls are in place to meet recovery time objectives and recovery point objectives.
3. The Corporate Audit group is responsible for confirming that business continuity and disaster recovery procedures are in place for the system and that the procedures comply with corporate standards.

Enforcement:
System Administrators or Data Owners who are found to be out of compliance with this policy will receive a lower performance rating for the year for failing the objective and may be subject to disciplinary action up to and including termination of employment.

Definitions:
1. Business Continuity Procedures:
    a. Business continuity procedures provide a list of actions as a response to an incident so that business operations may be resumed, ensuring the availability of critical assets.
2. Disaster Recovery Procedures
    a. Disaster Recovery procedures provide a list of actions as a response to a disruption or disaster that document the recovery of IT systems so that business operations may be re-established.
3. Recovery Point Objectives
    a. Recovery Point Objective is a measure of how much data loss can be tolerated (for example, 24 hours). Data back-up procedures should reflect the recovery point objective.
4. Recovery Time Objectives
    a. Recovery Time Objectives reflect how quickly the system needs to be recovered so that business operations are not critically impacted and is based upon business loss. (such as, the system must be recovered with 48 hours to resume regular processing or the impact will be $20,000 per hour in lost productivity).

Policy – Security Engineering in the Systems Development Life Cycle (section 10.1 of the ISO 17799 standard)

Purpose:
To establish guidance on security engineering in the Systems Development Life Cycle.

Scope:
This policy will apply to all information systems that are governed by the Systems Development Life Cycle contained in Company Policy (See Description of the System)

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Policy:
The company recognizes the importance of sound security practice and risk management within the Systems Development Life Cycle (SDLC). To ensure that security standards are practiced within the SDLC, the System Development Life Cycle team will establish checkpoint reviews and audit standards so that applications will meet a standard security baseline within the company. Development teams and their managers will review the security standards and practice checkpoint reviews over the course of development.

Roles:

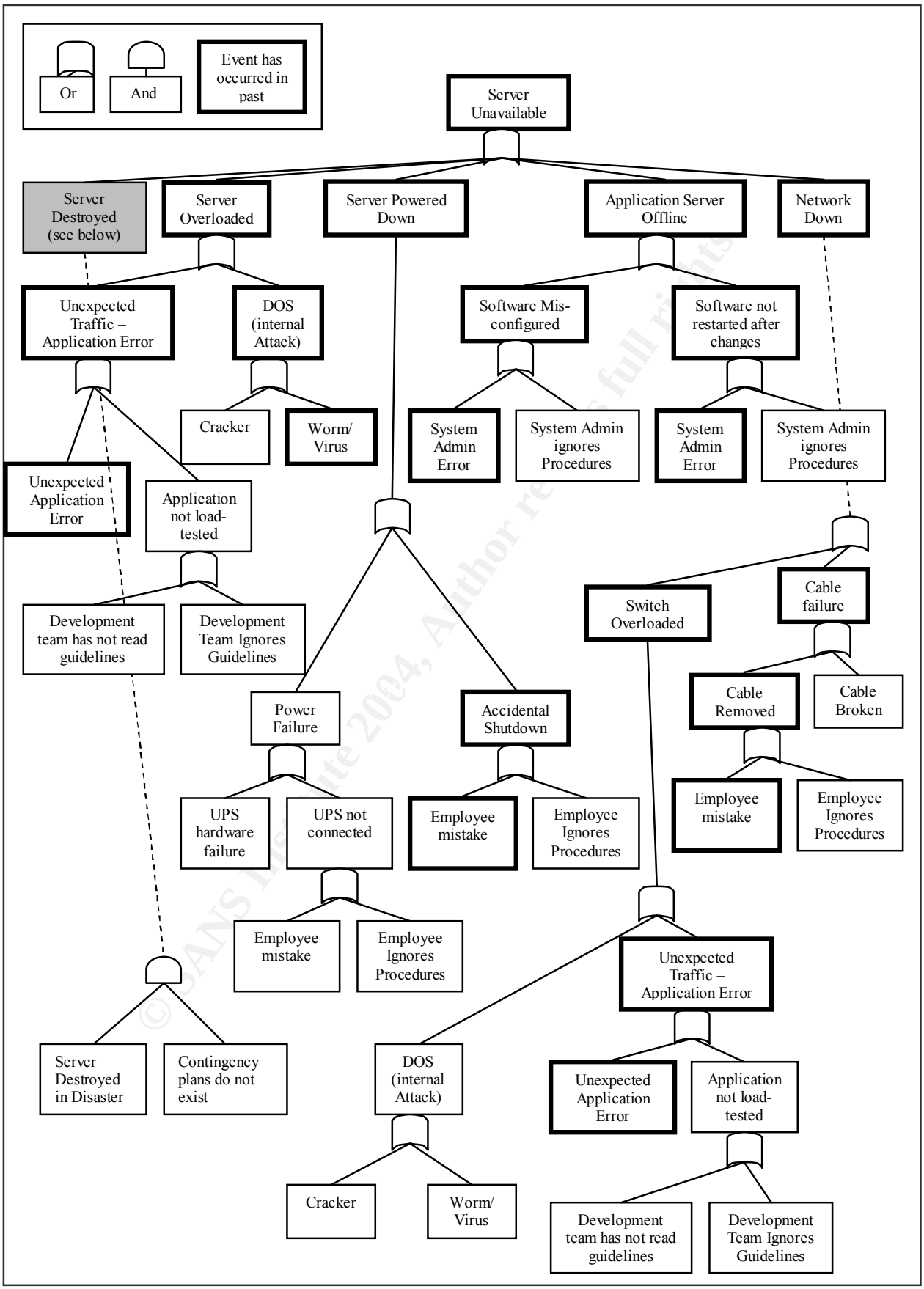
1. Developers are responsible for adhering to the standards during the development of systems.
2. Managers of Development Teams are responsible for ensuring that checkpoints are met before system deployment.
3. System Administrators are responsible for confirming checkpoint documentation prior to system deployment.
4. Business Data Owners are responsible for proper data classification and communication with the development team of risks that need to be controlled within the system.
5. System Development Life Cycle Team is responsible for establishing checkpoint reviews and audit standards. In conjunction with Corporate Information Security, the SDLC team will establish a set of baseline security standards.
6. Corporate Audit is responsible for auditing systems to ensure that they comply with corporate standards.

Enforcement:
Development teams who fail to comply with this policy will have their deployment of their application delayed until compliance is achieved.

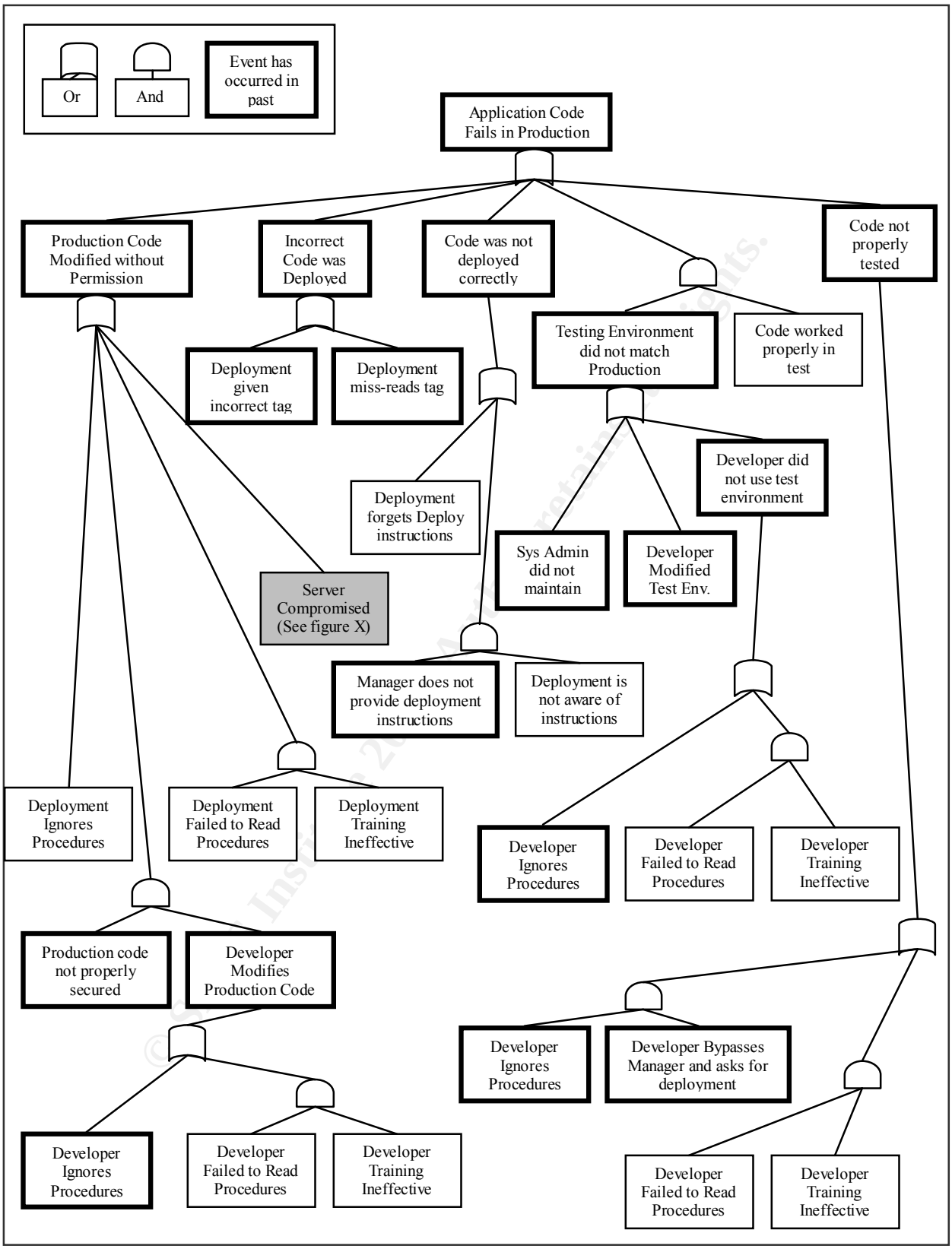
Definitions:
Systems Development Life Cycle (SDLC) is an engineering practice specified in Company Policy that provides a consistent systems development pattern so that applications meet a consistent engineering quality.

Appendix C – Fault Tree Analysis

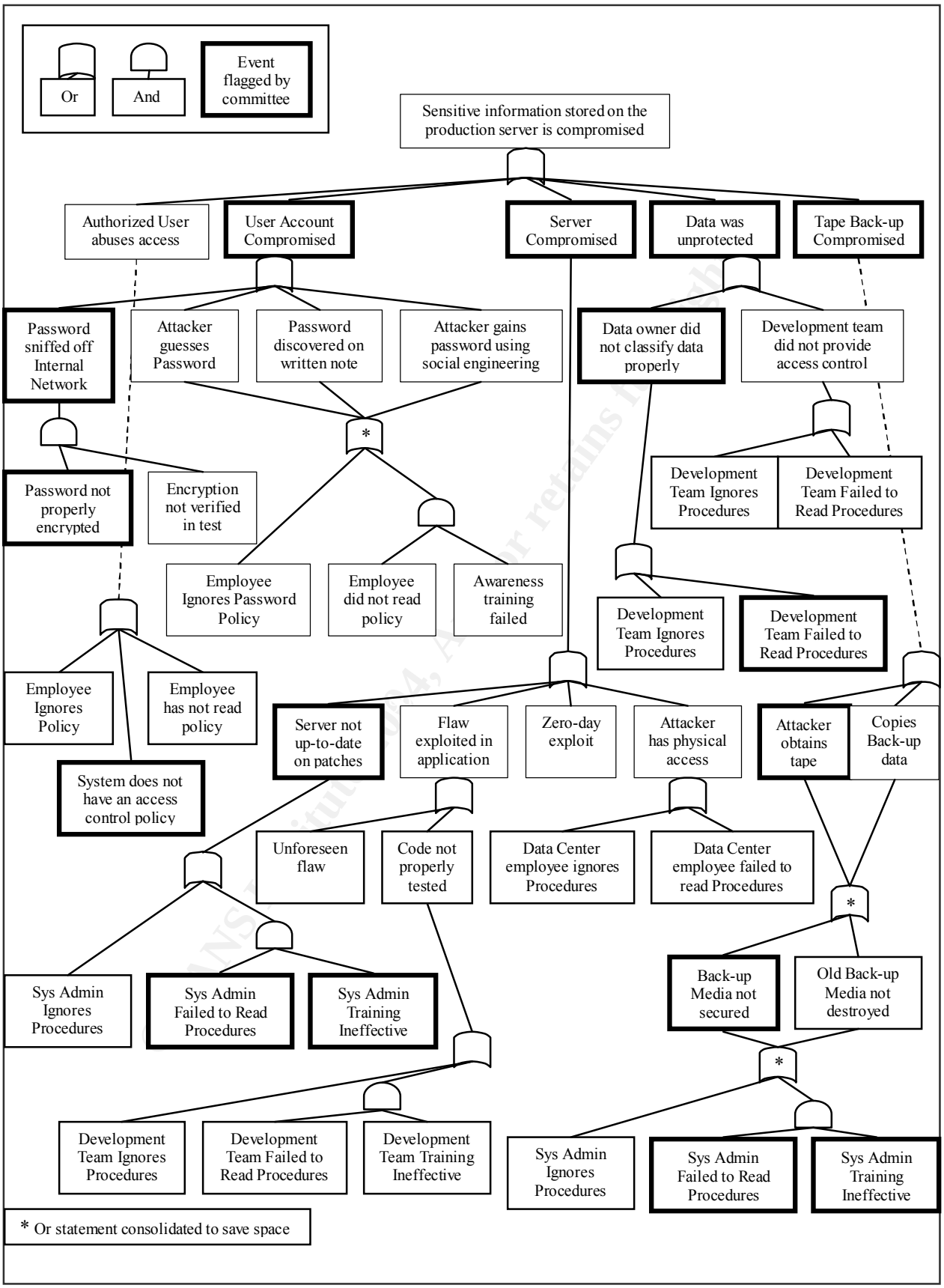
© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.



© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.



© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

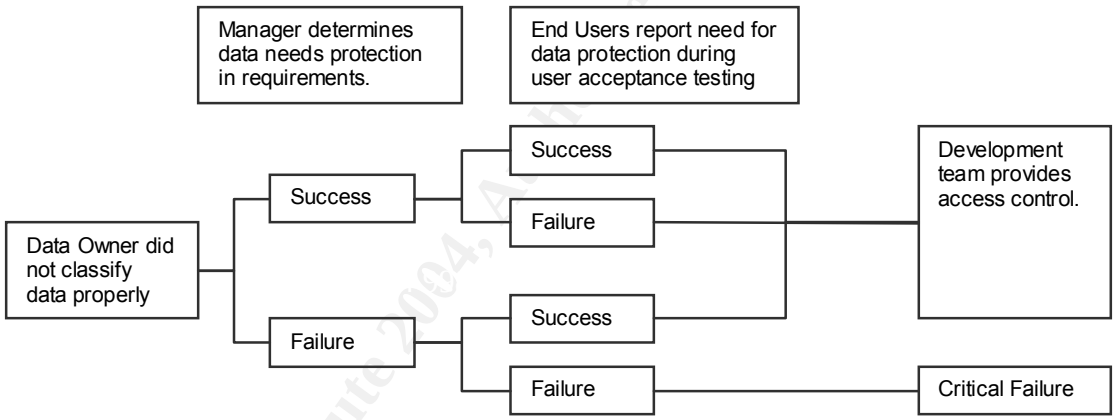


© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Appendix D – Flagged System Events

1. Disclosure of sensitive information due to improper data classification (data is unprotected) Description: Data Owners do not always classify their data because these are internal systems. Often they do not know that internal data is at risk to insider threats. These mistakes are often caught by the project manager in the requirements phase or by end users who feel that the data is incorrectly classified.
Frequency: 1 web site in 30.
Preventative Controls: The company has a policy of data classification and a widespread training program for all employees. Restricted information must have controls to limit user access.

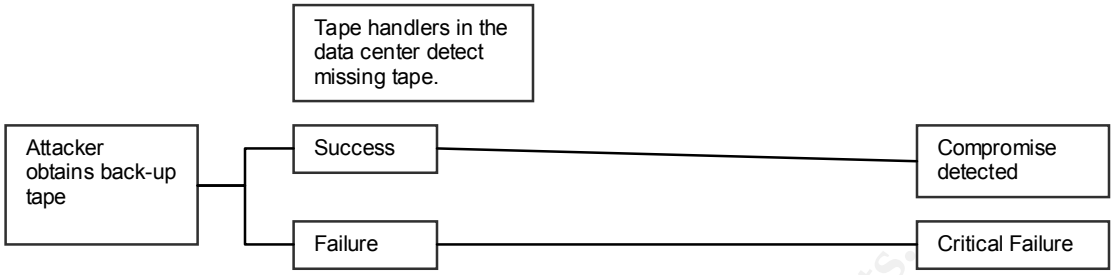
Event Tree



2. Sensitive information stolen from back-up media Description: During the Fault Tree Analysis, it was discovered that the system does not use encryption for back-ups. One of the data center personnel is in charge of swapping tapes on the system every week, but the system does not have specific tape handling procedures.
Frequency: Fault Tree Finding
Preventative Controls: Tapes are stored in a locked back room of the data center that is watched by tape operators. The data center is swipe card only and is manned 24 x 7. Visitors are required to sign in.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Event Tree



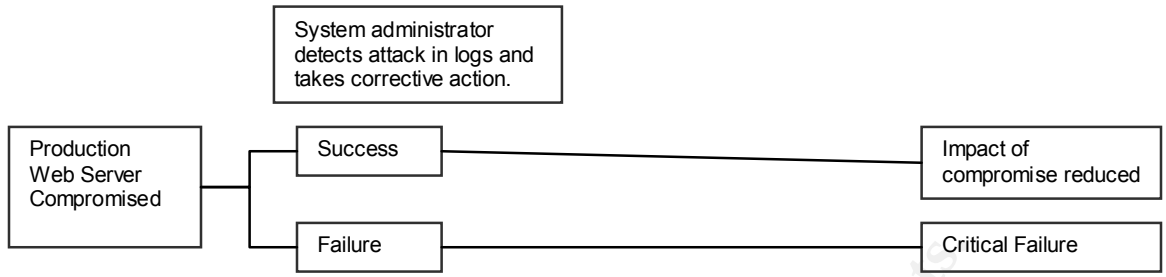
3. Authorized user abuses privileges and steals sensitive information **4. Contractor abuses system privileges and steals sensitive information** Description: The current system has no system of formal user registration/deregistration and no written statement of user rights and responsibilities. This will cause problems in reprimanding users who abuse their access privileges.
Frequency: Fault Tree Finding
Preventative Controls: None.
Detective or Reactive Controls: None

5. Use of Unencrypted channels (HTTP, FTP, Telnet) to convey sensitive information puts system at risk for internal packet sniffing. Description The examination of the information flow diagram found that most of the communications protocols used in the systems are not encrypted. Often, sensitive information, such as username and password is transmitted over these channels. This is illustrated in Figures 4 and 5 that show the protocols in the information flow diagrams.
Frequency: Most Channels, see Information Flow Diagrams (figures 4 and 5)
Preventative Controls: None.
Detective or Reactive Controls: None

6. Production Web Server compromised by internal hacker (using network) **7. CVS/NIS server compromised by internal hacker (using network)** Description The audits found that the Production Web Server is not regularly patched. This makes it vulnerable to internal attacks.
Frequency: Audit Finding
Preventative Controls: Server uses usernames and passwords to restrict access.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Event Tree



8. CVS/NIS or Production Web Server cannot be recovered after a disaster

Description:
Currently the system has no disaster recovery procedures.

Frequency: Fault Tree Finding

Preventative Controls: Currently, both servers have a full system back-up once per week. The back-up tapes are stored in a locked tape storage room in the data center. Both servers also have UPS support

Detective or Reactive Controls: None

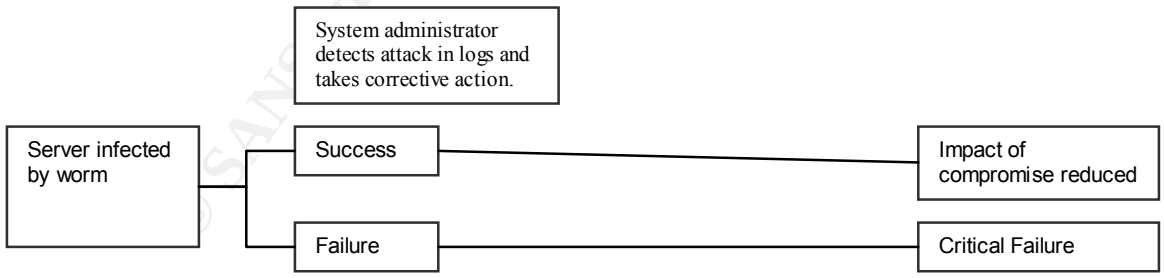
9. Production Web Server compromised by worm
10. CVS/NIS Server compromised by worm

Description
The audits found that the Production Web Server and CVS/NIS servers are not regularly patched. This makes them vulnerable to internal attacks from worms

Frequency: Audit Finding

Preventative Controls: None

Event Tree



11. Load Test Server compromised by internal hacker (using network)
12. Development Server compromised by internal hacker (using network)

Description
The audits found that the Load Test Servers and Development environment are not

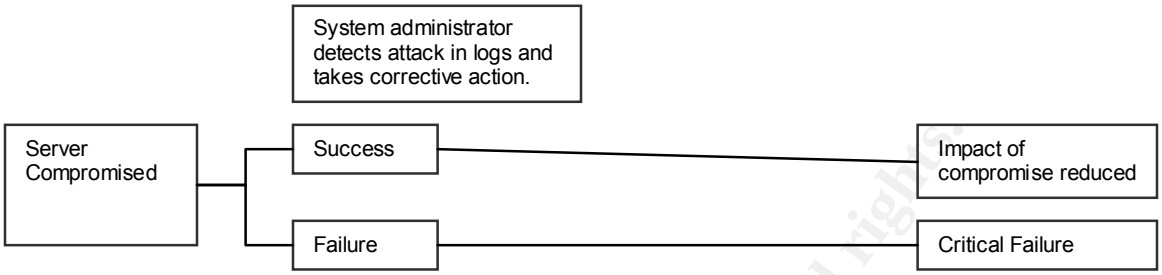
© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

regularly patched. This makes them vulnerable to internal attacks.

Frequency: Audit Finding

Preventative Controls: Server uses usernames and passwords to restrict access.

Event Tree



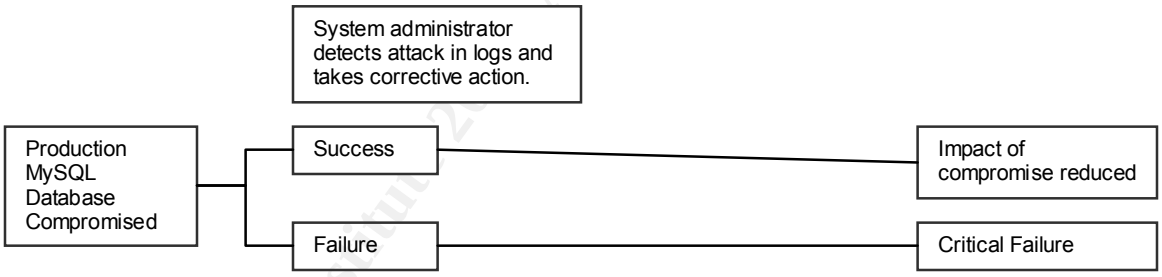
13. Information in Production MySQL Database compromised by hacker

Description
The audits found that the Production Web Server that hosts the MySQL database is not regularly patched. This makes it vulnerable to internal attacks.

Frequency: Audit Finding

Preventative Controls: Server uses usernames and passwords to restrict access. The database also uses usernames and passwords to restrict access.

Event Tree



14. Unauthorized modification of production code by developer

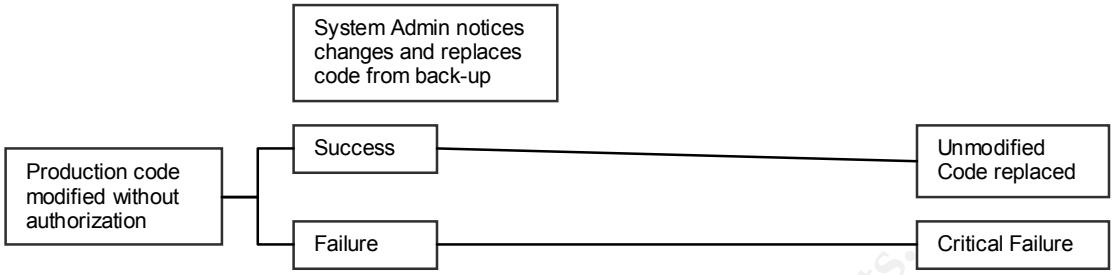
Description:
A couple of times, one of the developers has logged onto the production server and altered code in production because a client requested changes to the application. The developer was reprimanded for not following development procedures and the privileges for all developers were reduced to only read access. This problem represents a larger risk in that the system does not have automated integrity checkers.

Frequency: 1 time per year.

Preventative Controls: Privileges of the production code do not allow the developers to modify the code.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Event Tree



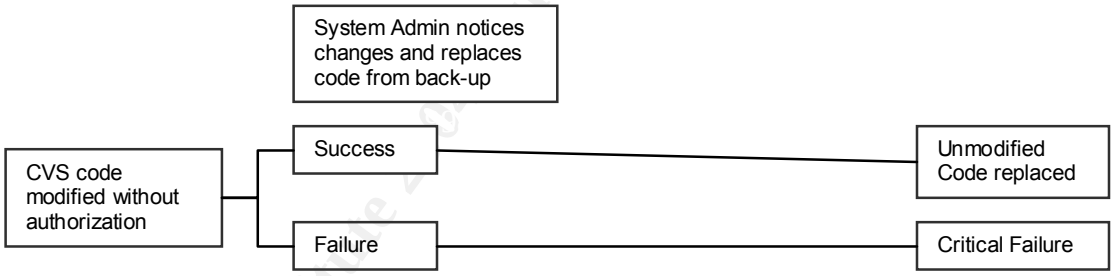
15. Developer accidentally deletes code directly on the CVS repository.

Description:
In general, developers only access CVS though the normal interface, making this unlikely. However, a developer could log onto the CVS/NIS server and modify the code. An accidental deletion code destroy development code

Frequency: Never Occurred.

Preventative Controls: Development procedures mandate that the developers modify CVS only via the CVS interface.

Event Tree



16. Network switch unavailable due to unintentional DOS

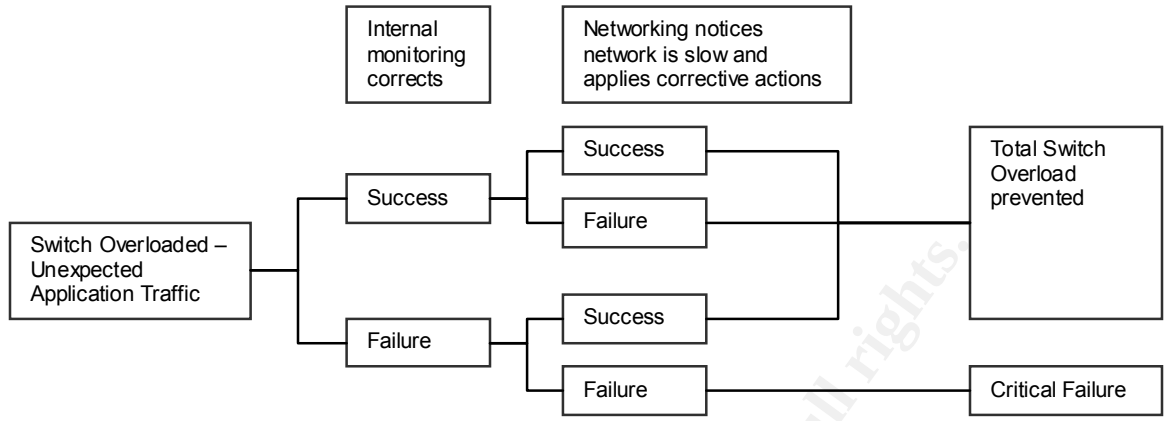
Description:
Last year, a switch became overloaded due to too much data being transferred. The switch was shutdown and restarted by networking who detected the problem. The network was down for a period of twenty minutes.

Frequency: 0.5 times per year

Preventative Controls: None

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Event Tree



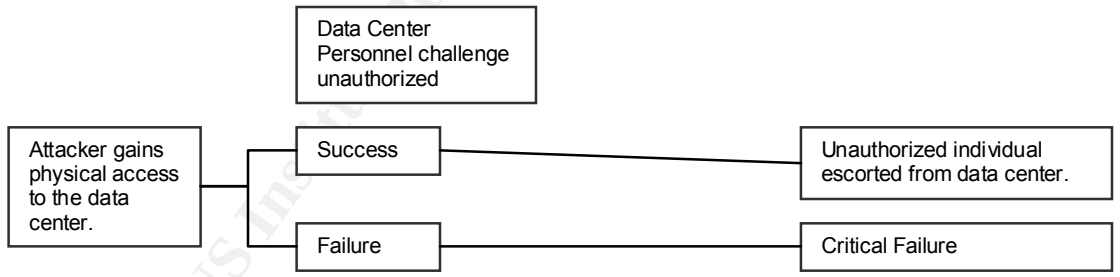
17. Attacker gains physical access to a server in data center.

Description
If an attacker can bypass the preventative controls of the data center, they can gain access to the Production Web Server, the CVS/NIS server or the Load Test Server.

Frequency: Fault Tree Finding

Preventative Controls: The data center occupies its own building. The data center is swipe card only and is manned 24 x 7. Visitors are required to sign in.

Event Tree



18. Attacker gains physical access to development server in development laboratory

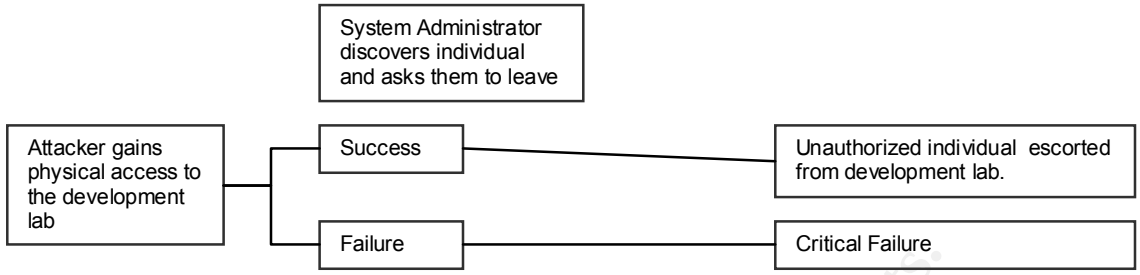
Description
If an attacker can get through the locked door in the development lab, they can gain access to the machines in the development environment.

Frequency: Fault Tree Finding

Preventative Controls: The development lab is a key-locked room.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Event Tree



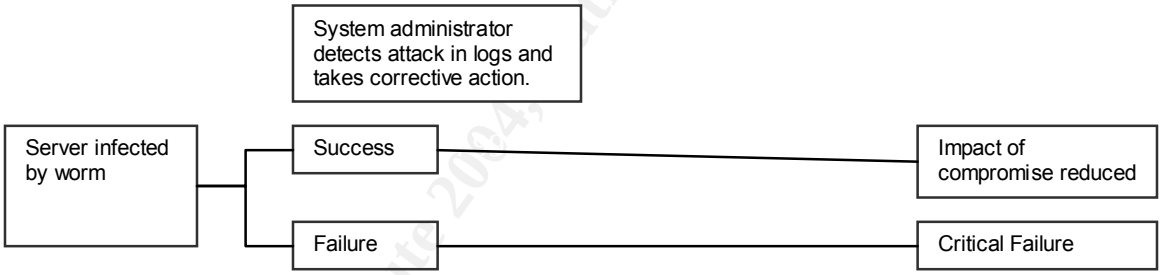
19. Load Test Server compromised by worm
20. Development Server compromised by worm

Description
The audits found that the Load Test Server and Development servers are not regularly patched. This makes them vulnerable to internal attacks from worms.

Frequency: Audit Finding

Preventative Controls: None

Event Tree



21. Information in production MySQL databases corrupted

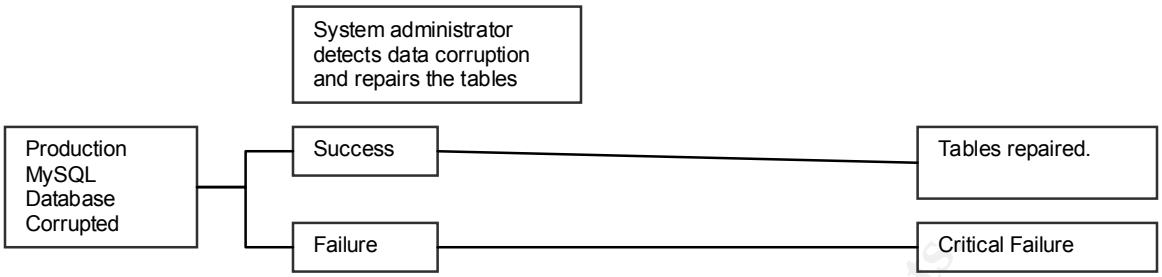
Description
A couple of applications have incorrectly passed data into the tables in the database, corrupting the data. In the past, the System administrator has caught the errors and has repaired the tables. In the past, this has resulted in minimal data loss (a few records).

Frequency: 2 times per year

Preventative Controls: None

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Event Tree



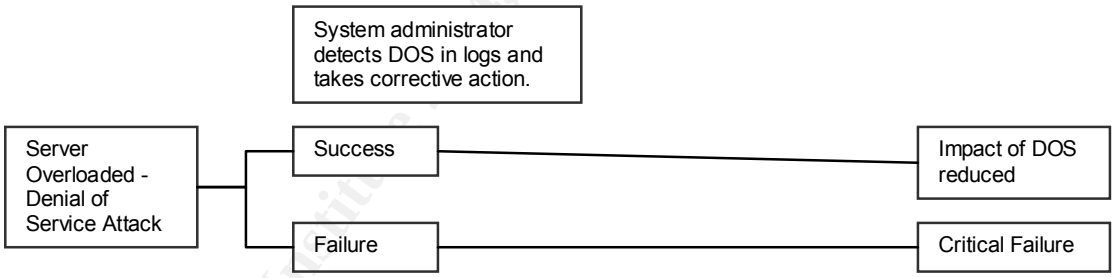
22. Server unavailable due to DOS attack (worm)

Description:
The company has had three major worm infections: Code Red infected the company twice and Blaster once. During the infections, several servers experienced denial of service attacks because of the number of requests from the worms. The impact was that network traffic was slowed down, but no server was completely shutdown. In past infections, the system administrator noted the attacks in the logs and reported them to the Corporate Information Security team who located the infected servers.

Frequency: 1.5 times per year

Preventative Controls: None

Event Tree



23. Server unavailable due to unintentional DOS (application error)

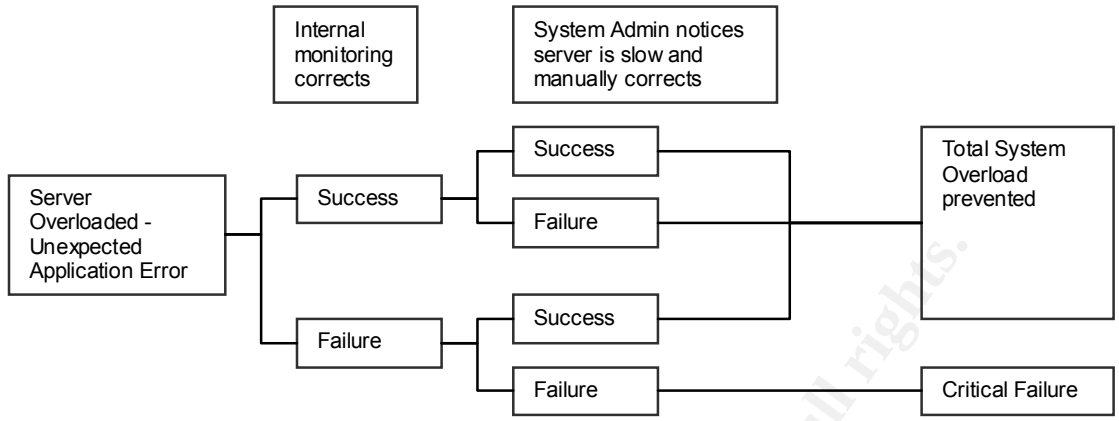
Description:
The application which tracks patches and deployments in the desktop environment sometimes becomes overloaded with desktops transmitting information to the server. The system administrator has written scripts to monitor for an overload in this application and kill off the processes during an overload.

Frequency: 3 times per year

Preventative Controls: Most applications that could overload the server are load tested prior to production.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Event Tree



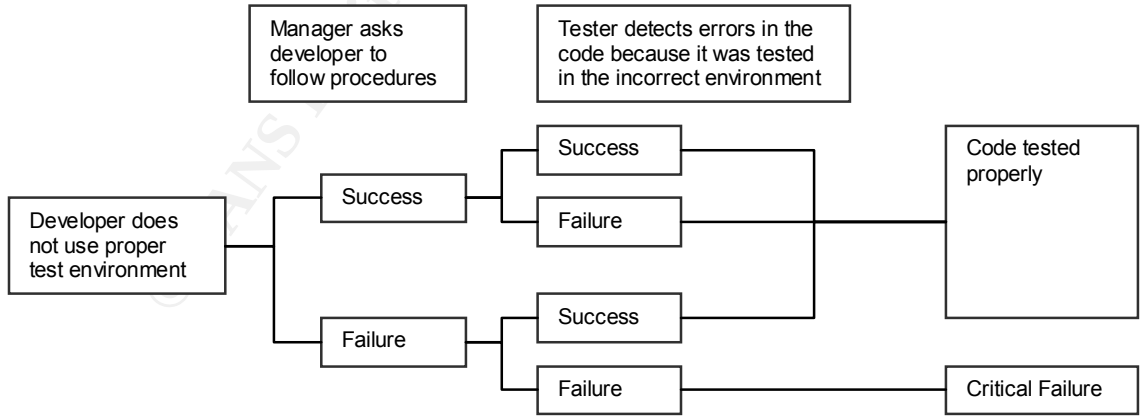
24. Loss of application availability due improper testing

Description:
Developers often install a web server environment upon their laptop unit to more rapidly develop code. Some developers will bypass normal test procedures and not test the code within the development environment. Often, these problems are caught by the manager or tester in testing phases. Problems occur when the application is behind schedule and rushed from development to production.

Frequency: 10 times per year

Preventative Controls: The team has development procedures that mandate testing in the proper environment.

Event Tree



25. Production Web Server unavailable due accidental shut-down

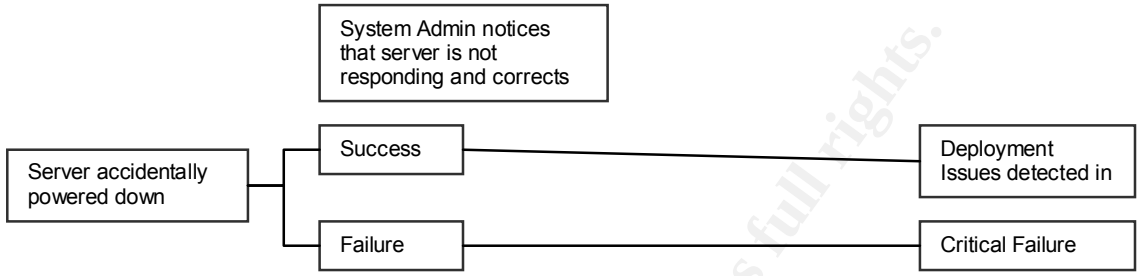
© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Description:
A couple of times, an administrator has accidentally powered down the server during system administration duties. This has always been rapidly detected by the system administrator because the server was not responding.

Frequency: 1 time per year.

Preventative Controls: None

Event Tree



26. Application server unavailable due to misconfiguration

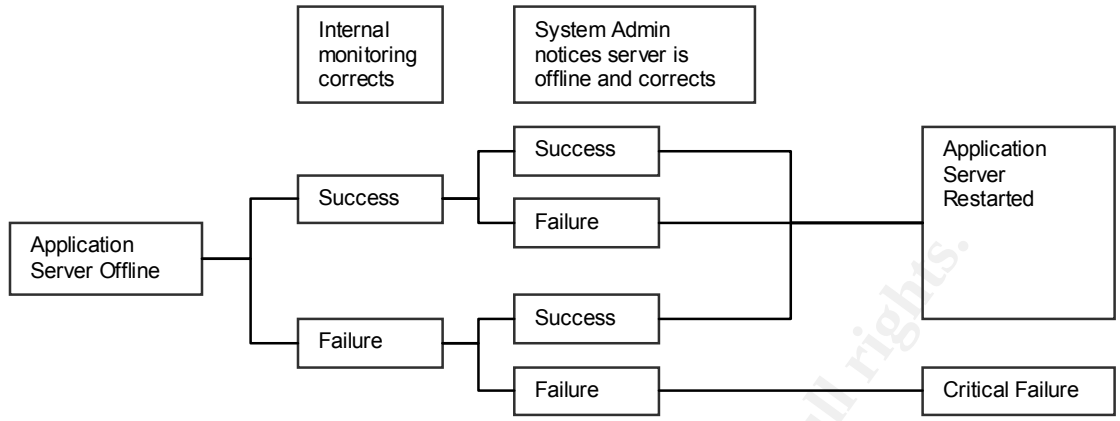
Description:
During a deployment the system administrator needs to restart the java web server to add new applications into context, so that the web server begins to display the content. At times, the System Administrator has forgotten to restart the web server or has misconfigured the server, so that it does not restart properly. The system administrator has written internal scripts that monitor the status of the web server and attempt to restart the web server if it has been offline too long. In addition, the system administrator typically confirms that the server is active after a deployment, so this fault is rapidly corrected.

Frequency: 2 times per year

Preventative Controls: There are unwritten procedures in place to test changes of configuration in development prior to deployment. There are also unwritten procedures to confirm that the application server is functional after a deployment.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Event Tree



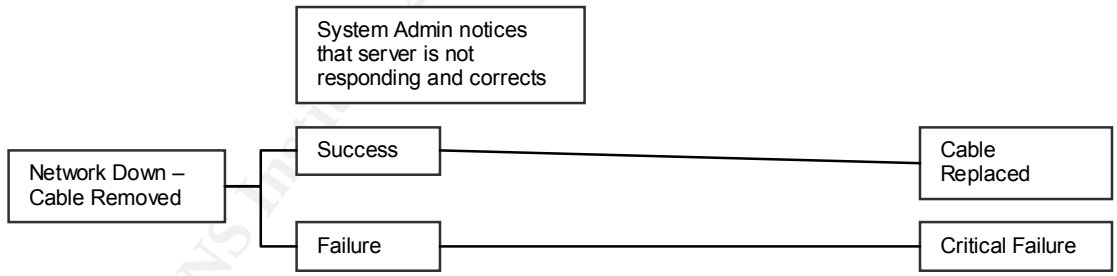
27. Development server unavailable due to missing network cable

Description:
In the development environment, developers will sometimes move machines and unplug network cables. This has resulted in development machines being unavailable, but has been more of an annoyance than a problem. This has never occurred in the production environment.

Frequency: 1.5 times per year

Preventative Controls: None

Event Tree



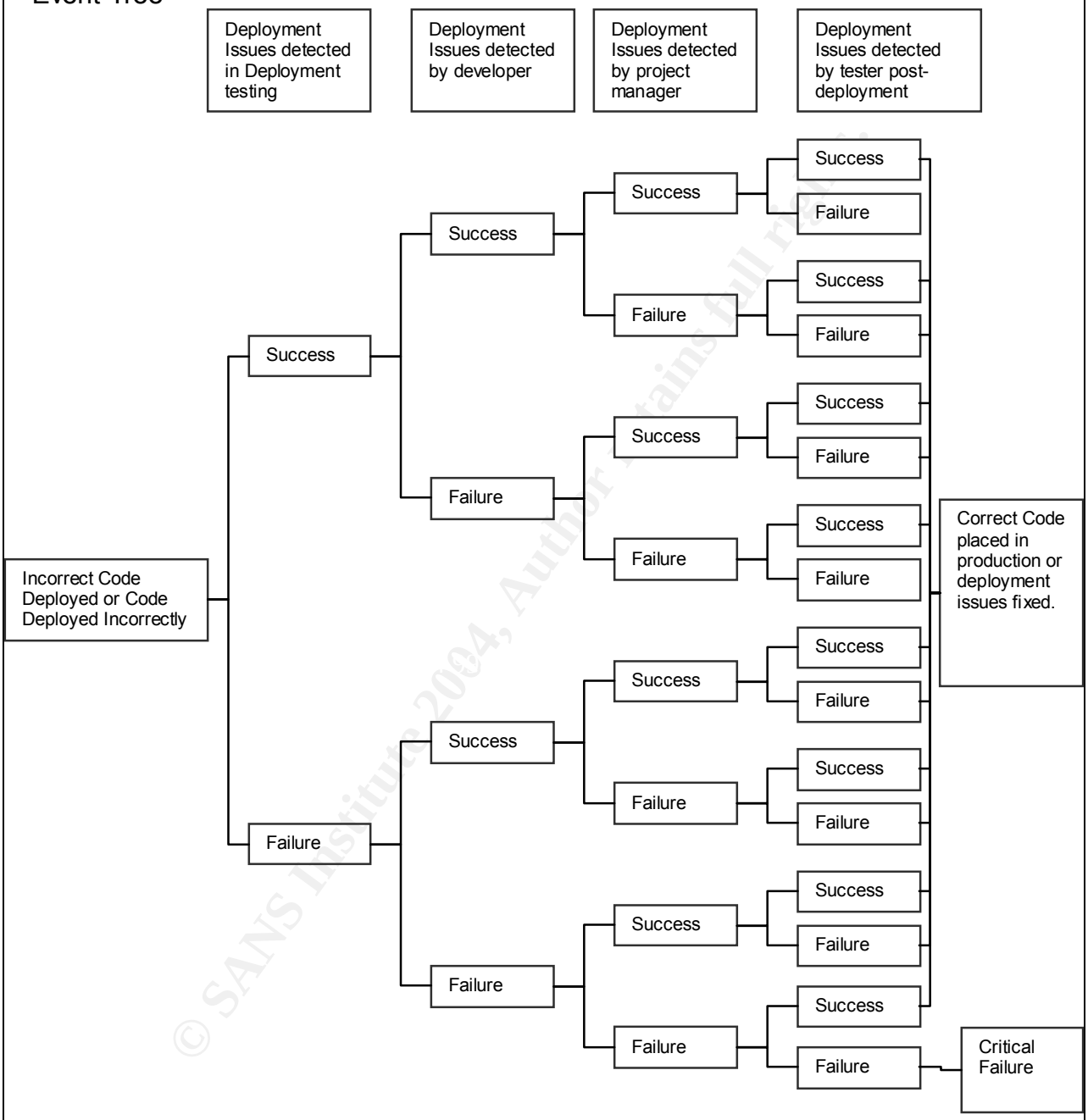
28. Incorrect Code Deployed or Code Deployed Incorrectly, resulting in code that does not operate correctly in production.

Description:
Some applications have special deployment instructions so that the code will work properly. Sometimes, the project manager forgets to pass the instructions to the system administrator. In addition, sometimes the developer passes the wrong tag to system administrator for deployment. Typically, the result is that the code does function properly. Since the managers, developers, testers and system administrator often test

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

the code after deployment, this issue is rapidly caught and corrected.
Frequency: 2 times per year
Preventative Controls: None

Event Tree



© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Appendix E – High Level Plan for Risk Management

Key	
Priority:	H = high
	M = medium
	L = low
Controls	X = Control in place
	N = Control needed

	Priority	Policy	Procedure /Standards	Patches	Encryption	Log Watcher	Anti-Virus	IDS	Integrity Checker
1.Disclosure of sensitive information due to improper data classification (data is unprotected)	H	X	N						
2. Sensitive information stolen from back-up media	H	X	N		N				
3. Authorized user abuses privileges and steals sensitive information	H	N	N			N			
4. Contractor abuses system privileges and steals sensitive information	H	N	N			N			
5. Packet Sniffing on internal network	H	X	N		N				
6. Production Web Server compromised by internal hacker	H		N	N				N	N
7. CVS/NIS Server compromised by hacker	H		N	N				N	N
8. CVS/NIS or Production Web Server cannot be recovered after a disaster	H	N	N						
9. Production Web Server compromised by worm	M		X	N			N	N	
10. CVS/NIS Server Compromised by worm	M		N	N			N	N	
11. Load Test Server compromised by hacker	M		N	N				N	N
12. Development Server compromised by hacker	M		N	N				N	N
13. Information in Production MySQL Database compromised by hacker	M		N	N				N	N
14. Unauthorized modification of production code by developer	M		N						N

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

15.Integrity of source code in CVS compromised	M		N			N			N
16. Network switch unavailable due to unintentional DOS (application error)	M		X					X	
17. Attacker gains physical access to production server in data center	L	X	X			N			
18. Attacker gains physical access to development server in development laboratory	L	X	X						
19. Load Test Server Compromised by worm	L		N	N			N	N	
20. Development Server compromised by worm	L		N	N			N	N	
21. Information in production MySQL databases corrupted	L		N						
22. Server unavailable due to DOS attack (worm)	L		N					N	
23. Server unavailable due to unintentional DOS (application error)	L		N			X			
24. Loss of application availability due improper testing	L		X						
25. Production Web Server unavailable due accidental shut-down	L		N			N			
26. Application server unavailable due to misconfiguration	L		N						
27. Development server unavailable due to missing network cable	L		X						
28. Incorrect Code Deployed or Code Deployed Incorrectly, resulting in code that does not operate correctly in production.	L		N						

© SANS Institute 2004, Author retains full rights.

Appendix F – Extended Audit Checklist

Incident Management Procedures (ISO 17799: 8.1.3)
Objective: In the past, the company has experienced incidents of internet worms, resulting in denial of service. The purpose of the incident handling procedures is to provide a timely response to security incidents and to contain and correct the incident to prevent loss for

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

the company. In particular, at minimum, the system must possess handling procedures for:

1. system compromise
2. denial of service (DOS) attacks
3. unauthorized disclosure of sensitive information

The incident handling procedures should be regularly reviewed and updated to reflect changing threats and risks.

Audit steps to determine compliance:

1. Identify the incident handling documentation for the system.
2. Check the last review/modification date of the procedures.
3. Verify that there is an owner of the procedures to review and update them.
4. Verify that the system administrator and incident handling team has read and is aware of the procedures.
5. Verify that the development team is aware of the procedures.
6. Verify that the procedures contain sections for handling:
    a. system compromise
    b. denial of service (DOS) attacks
    c. unauthorized disclosure of sensitive information
7. Verify with Corporate Information Security that incidents arising from the system have been handled according to the procedures.
8. Verify that the procedures have been modified according to lessons learned from the incidents.

Business Continuity Management (ISO 17799: 11.1)

Objective:
The company considers the ISMS to be an important asset in business operations. To ensure continuing operation of the system, the ISMS must have procedures for restoration of the system in the event of a disaster. This plan should be commiserate with asset loss values and must be regularly tested and updated to ensure effectiveness.

Audit steps to determine compliance:

1. Check that the asset inventory is up to date.
2. Identify the system disaster recovery procedures
3. Check the last review/modification date of the procedures.
4. Verify that there is an owner of the procedures to review and update them.
5. Verify that the system administrator has read and is aware of the procedures.
6. Verify that a back-up tape yields recoverable data (see Systems back-up auditing)
7. Verify that the systems administrator has documentation verifying that the disaster recovery plans have been tested in the past year.

Operational Procedures and Responsibilities (ISO 17799: 8.1)

Objective:

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

To ensure proper operation of the ISMS, standard operating procedures will be documented and used to maintain the system. At minimum, these procedures will contain sections for:

1. System Back-Up
2. Change Control
3. Patching
4. Proper Shutdown
5. Deployment of Production Code

These procedures will be updated to reflect changes in the system.

Audit steps to determine compliance:

1. Identify the system procedures that document :
   a. System Back-Up
   b. Change Control
   c. Patching
   d. Proper Shutdown
   e. Deployment of Production Code
2. Check the last review/modification date of the procedures.
3. Verify that there is an owner of the procedures to review and update them.
4. Verify that the system administrator has read and is aware of the procedures.
5. System Back-Up
   f. Verify that the back-up tapes are present and being rotated properly.
   g. NOTE: The following procedures are also used (in part) to verify that the back-up tapes are being encrypted.
   h. Using the java tools and escrow key files, decrypt the tar file from back-up
   i. Untar the back-up files and verify that the files are intact.
   j. Verify that 1 weeks worth of back-up content files from the production web server and the CVS server are present on the Central logging server (using ls –l).
   k. Untar (tar – xvf <filename>) one of the content files form CVS and a second on from production to verify that the back-ups are being captured.
6. Change Control
   l. Verify that proper change control forms exist.
   m. Review the last set of changes with the system administrator.
7. Patching
   n. Check recent vulnerability announcements and select vulnerabilities that would need to be patched on the system.
   o. Verify that that change control forms have been filled out to patch these vulnerabilities.
   p. Run nessus against the system to verify that the system is up to date on patches.
8. Proper Shutdown
   q. Verify with web development managers that there have been no accidental shutdowns reported.
   r. Verify shutdowns in server history logs and syslogs.
9. Deployment of Production Code
   s. Interview web development managers and verify that code has been

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

deployed correctly (there have been no complaints).

Standards for Cryptographic Controls **(supports the Policy on use of cryptographic controls [ISO 17799: 10.3.1])**
Objective: To protect encrypted sensitive information, the ISMS will have published standards for encryption that are regularly updated to reflect changing trends in cryptography and cryptographic algorithms. These standards will support the company encryption policy.
Audit steps to determine compliance: 1. Identify that the encryption standards 2. Verify that the standards are published 3. Verify that the standards have and owner responsible for regular review. 4. Check the last modification/review date of the standards. 5. Check that the Systems Operations team have reviewed the standards. 6. Check with the Systems Operations team that the standards have been implemented in the selection of cryptographic controls.

Cryptographic Controls for Sensitive Information **(Covers Network Controls [ISO 17799: 8.5.1] and Security of Media in Transit [ISO 17799: 8.7.2])**
Objective: To prevent the loss of proprietary information, cryptographic controls will be implemented that protect the confidentiality of information transmitted over the network and stored onto removable media.
Audit steps to determine compliance: Network Controls 1. Verify with the system administrator that cryptographic controls have been implemented for security of network communications. 2. Using nmap (nmap <target ip>) verify the TCP services running on each system. 3. Using nmap (nmap –sU <target ip>) verify the UDP services running on each system. 4. Review the services running on each system and verify with the Systems Operations Committee that each service is necessary for business operations. 5. Document the services that are running unencrypted. Using amap, verify the active services (by banner grabbing). 6. If given written permission by the Information Management Steering Committee, it is suggested that ettercap be used to test for unencrypted information flowing over the switch. Tape Security 7. Verify with the system administrator that cryptographic controls have been implemented for security of media in transit. 8. Extract the back-up tar file from tape. Using tar, attempt to extract the file (tar –xvf <filename> - this should fail because the file is encrypted and not recognized properly by tar). 9. Using the java tools and escrow key files, decrypt the tar file.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

10. Untar the back-up files and verify that the files are intact.

User Access Management (ISO 17799: 9.2) [condensed]

Objective:
To reduce the risk of proprietary information loss, the system will have procedures for

5. formal user registration and de-registration.
6. allocation of system privileges.
7. handling and use of user credentials, such as passwords.
8. the periodic review of access rights.

A user awareness program will be established so that users are aware of their rights and responsibilities. The system will be periodically audited to ensure that procedures are being implemented upon the systems.

Audit steps to determine compliance:

2. Identify the system procedures that document :
   a. formal user registration and de-registration.
   b. allocation of system privileges.
   c. handling and use of user credentials, such as passwords.
   d. the periodic review of access rights.
3. Check the last review/modification date of the procedures.
4. Verify that there is an owner of the procedures to review and update them.
5. Verify that end-users are aware of their rights and responsibilities.
6. Examine the /etc/passwd file (less /etc/passwd) and verify that all user accounts are current company employees. On production, verify that none of the accounts are contractors.
7. Examine /etc/passwd and verify there are no shared accounts.
8. Examine /etc/shadow (less /etc/shadow) and verify that all accounts have passwords.
9. In /etc/shadow, verify that root passwords must expire in 30 days
   e. The root password accounts should look like:
   f. <username>:<password>:<last changed>:<may change>:30:<warn>:<disable>:<expire date>:
10. In /etc/shadow, verify that other passwords must expire in 90 days
   g. The other password accounts should look like:
   h. <username>:<password>:<last changed>:<may change>:90:<warn>:<disable>:<expire date>:
11. If given written permission by the Information Management Steering Committee, it is suggested that a dictionary attack be performed on the password file to verify that no dictionary words are present (using John the Ripper).

Development Checkpoints for the SDLC
Covers:
Security Requirements analysis and specification (ISO 17799: 10.1.1)
Change Control Procedures (ISO 17799: 10.5.1)

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Technical Review of operations systems changes (ISO 17799: 10.5.2)
Objective: To reduce the risk of proprietary information loss through incorrectly classified information the system shall have development checkpoints to confirm that access control is implemented for sensitive assets. In addition, development checkpoints will be established to ensure proper operation of production code. The development procedures will be reviewed on a yearly basis.
Audit steps to determine compliance: 1. Verify that development procedures are documented. 2. Verify that there is an owner of the procedures to review and update them. 3. Verify that development procedures have been reviewed in the last year. 4. Check that developers, managers and content owners have are aware of the procedures. 5. Check that developers and managers have read the procedures. 6. Review web site content and verify that it is classified with the correct labels. 7. Verify with content owners that the information is correctly labeled by distributing a form in which they document the classification of content on their web site. 8. Verify the web site contains no publicly available content labeled restrictive or highly-restrictive.

Monitoring System Access and Use (ISO 17799: 9.7)
Objective: The system shall maintain system logs that document the system access and use. The logs will be periodically reviewed for evidence of system abuse. The system will maintain logs of system events and the logs will be audited for evidence of system abuse using automated tools. System abuse will be reported according to incident handling procedures to the incident handling team
Audit steps to determine compliance: 1. Verify with the systems administrator that the system maintains logs and that the logs are periodically reviewed. 2. Portscan the machine with Nmap. 3. Verify alerts in the syslog. 4. Verify that the alert is e-mailed to the system administrator. 5. Using Nessus, do a full system scan. 6. Verify alerts in the syslog. 7. Verify that the alerts are e-mailed to the system administrator. 8. Check that the incident handling team has procedures in place to handle incidents of system abuse. 9. Check that past incidents have been filed with the incident handling team and review resolutions of the incidents.

Security of System Files (ISO 17799: 10.4)
Objective: To ensure the integrity of system files, monitoring tools will be implemented that make daily scans and generate alerts to changes in system files. The alerts that are not associated with normal system operations (such as a deployment) will be handled

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

according to incident handling procedures and reported to the incident handling team.
Audit steps to determine compliance: 1. Verify with the systems administrator that integrity checking software has been installed on the system and is operating. 2. Back-up the /etc/passwd file. Add a new account to /etc/password 3. Verify the alert in the syslog. 4. Restore /etc/passwd. 5. Back-up the index page of the web content server. 6. Insert a comment (<!—inserted comment -->) into the index page. 7. Verify the alert in the systlog. 8. Check that the incident handling team is aware of the integrity alerts and that incident handling processes are in place.

Protection against malicious software (ISO 17799: 8.3)
Objective: To prevent compromise of the servers anti-virus software will be implemented that makes daily scans and generates an alert in the system log and will proceed to eradicate the malware. To be effective, the anti-virus signatures must be updated daily.
Audit steps to determine compliance: 1. Verify with the system administrator that anti-virus software has been installed on the system and that signatures are updated daily. 2. Install an eicar file (link) and examine the system log to determine if the antivirus software has correctly identified the test file.

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.