



---

## White Paper

---

---

# Information Security Risk Management: A Systems Engineering Process

---

---

### Disclaimer

This is one of a series of articles detailing information security procedures as followed by the INFOSEC group of Computer Technology Associates, Incorporated, also known as CTA. These articles are copyright by Computer Technology Associates and may not be reproduced or used for profit without the expressed written permission of CTA or as included in contractual arrangements with clients of CTA.

For further details as to the process and the procedures followed, contact:

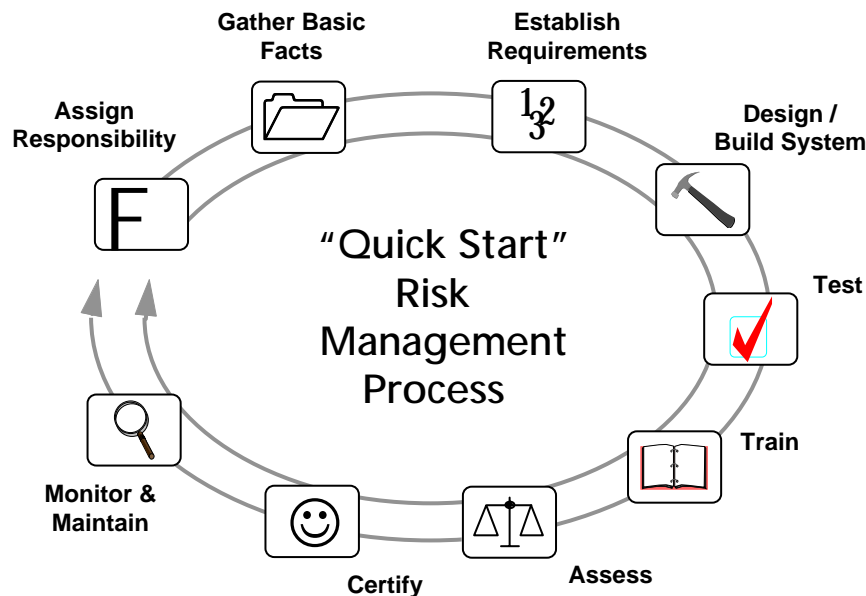
Computer Technology Associates, Inc.  
INFOSEC Group  
7150 Campus Drive, Suite 100  
Colorado Springs, CO 80920  
(719) 590-5100

## Overview

Information Security Risk Management seeks to balance an organization’s expenditures on security measures and controls against the business value of the assets and business activities at risk if the enterprise’s security policy is compromised. Our Risk Management process has nine steps designed to ensure that an acceptable level of information security risk is achieved and maintained during the entire life cycle of mission critical systems. The process is applicable to new and existing systems and is usually tailored to accommodate special client situations.

Principles of systems engineering (see related white paper “Systems Engineering Methodology”) have been extended to define our systems security engineering discipline. The key to the process is a rigorous development of security functional, performance and operational requirements and the subsequent decomposition and allocation of these requirements to a system’s architectural components. Each process step described below has a specific purpose. Achieving this purpose is more important than the process step’s “paperwork” or “formats”. If the step in question does not quite fit the situation, achieve the objective of the step by tailoring the process.

The nine steps to the process are:



Note: The process is documented using the Risk Management Worksheet at Attachment 1. Each step will provide general instructions to guide your completion of the worksheet. Since the worksheet is generic to all systems use them as a guide and descriptive of the objective of the process step; customize, add attachments and/or references as necessary to reflect the specific environment being analyzed. Assuming the enterprise IT infrastructure is comprised of independent but interrelated systems, each system should have its own worksheet.

Documentation for the process has been streamlined for ease of implementation across your enterprise. In several of the steps, the advantages of more strenuous documentation than that required by the Worksheet are suggested. The paragraphs that follow will discuss each of the basic steps and some suggested extensions.

## Step 1 - Assign Responsibility

Fundamental to successful information security management is a policy that establishes security management guidelines as an integral part of the organization's overall business strategy. This policy, in turn, drives the development of an overall security management architecture that balances cost and risk for a new or existing system. A single point of contact for the system's security should be identified who will be responsible for the development of the system's security management architecture and be an active contributor in all stages of the system's life cycle.

Assign security responsibility for existing systems if a single point of contact for information security doesn't exist...the rule "better late than never" applies to this and all other steps in the process.

### Action required to complete Step 1

Complete the pertinent blocks in the "**General Information**" section of the Risk Management Worksheet.

## Step 2 - Gather Basic Facts

It's embarrassing not to be able to provide basic facts about the information on your system. Simple questions like, "how sensitive is the information on your system", "who owns the information", or "where does it go", are really tough to answer if you haven't given it some thought. Step 2 provides the opportunity to think about and document basic facts about the system and its information. The step is applicable to existing, migrating, or developing systems alike. It is well established that very few enterprises know what information resides on their networks, where it is located, who has access to that information and the cost of having that information compromised. This lack of self-knowledge becomes a crippling limitation when it comes time to convert generalized policy principals into specific security mechanisms, generally resulting in the over-protection of assets of marginal value, and/or the under-protection of critical and/or highly sensitive assets.

The way you answer the Worksheet questions will determine the rigor you apply to your system's security, ensuring an appropriate match of safeguards and asset value and/or sensitivity.

### Sensitivity and Criticality

Assess both the sensitivity of the data input, processed, stored, and transmitted by the system and the system's criticality to operations. The sensitivity assessment should also consider personal customer information that if compromised, could result in civil penalties and/or loss of customer confidence. This is an important step since the results of this assessment will dictate the level of protection afforded to the system. To get a complete and accurate representation of system data sensitivity and criticality, it is useful to examine each category of data (e.g., Company Proprietary, Employee Private, Customer Private, etc. as summarized in part 2a. of the Worksheet) against the consequences of 3 basic vulnerabilities: breach of **confidentiality** (e.g., if data is released), loss of **integrity** (e.g., if data is maliciously modified), or denial of **availability** (e.g., if service is denied or data destroyed).

This system-level analysis will generally reveal important data distribution issues such as: System criticality level 3 data may be distributed across several applications and system platforms, and/or Level 0 information may co-reside with higher level information on the same platform. Understanding such issues may be crucial to the success of the design approach taken to secure the higher-level assets. For example, if the situation is a new system development, allocating all higher level functions and data to a logically and physically isolated platform may be preferable to tackling a multi-level security solution on a single platform.

## ☐ Threats

Assess the possible threats to the system. Table 1 provides examples of threats for each of the four threat types. It is important to factor the **likelihood** of the threat occurring and the **impact** if it does into the assessment. The importance of focusing on real threats rather than theoretical ones cannot be overstated. On the other hand, prudent risk management demands preparing for disaster recovery if the impact of loss would be catastrophic, even due to an unlikely, but possible threat (e.g. natural disaster).

In this post September 11 environment, of biggest concern to many enterprises is the threat of organized cyber terrorism. The recent growth in human intentional attacks on the nation's high-profile computer systems has resulted in the establishment of several Federal agencies that publish up-to-date statistics on experienced cyber attacks and suggested actions to thwart such attacks. This data can be particularly useful in establishing a baseline set of security management requirements leveraging the cooperative research activities of these agencies.

**Table 1. Threat Types and Examples.**

Threat Type:	Examples:
Human Intentional	Malicious intruder (hacker) Terrorism / attack Corporate espionage / competition Disregard for procedures Disgruntled employee
Human Unintentional	Curiosity seeker Untrained user Data entry error Programming or configuration error
Structural	Physical environment Hardware anomaly Software anomaly Power anomaly
Natural	Fire Wind Flood

## ☐ Other

Other important facts to be gathered during this stage include existing and planned interfaces to the system (both internal to the company and external to the company) and information about the user community or communities (again internal and external). This information is crucial to understanding a system's vulnerabilities, since open networks are like chains in which the system with the weakest security creates a threat to all other systems on the same network.

## ☐ Action required to complete Step 2

Complete the "**Sensitivity and Criticality Information**", "**Threat Information**", and "**User Community Information**" sections of the Worksheet.

## Step 3 - Establish Requirements

The security systems engineering process begins with the development of complete, consistent and testable requirements, followed by a decomposition and allocation of requirements to a security systems architecture and finally designed into the components of the target system: operational processes, hardware and software components.

Establishing **functional, performance and operational** security requirements is just as important to a meaningful risk management process as it is to a systems development process. Information security **functional** requirements include specific confidentiality, integrity and availability requirements of a system (the “what” of security requirements); **performance** requirements specify the quantitative aspects of these requirements (e.g. ability to thwart a specific threat scenario, detect and respond to a specific attack within a specific timeframe, etc.) and **operational** (procedural) requirements specify the “who” and “how” system functions will be used (e.g. user identification/authentication, access controls, etc.). Operational requirements also specify physical controls in the system’s environment. For example, operational security requirements might be that the system will lock out its user’s access after 15 minutes of inactivity, and that the terminal will only be accessible via a “smart card” behind a cipher locked door, etc.

For existing systems, security requirements may result from a formal vulnerability analysis on the system and evolving threats. For example, automated network and host scanning tools are available to assess and identify vulnerabilities in network services, architecture, operating systems and applications associated with vendor-supplied software (e.g., missing patches, poor default configurations, etc.), system administration (e.g., poor password management, improper firewall configuration, etc.) and/or user activity (e.g., policy avoidance, failure to update virus software, etc.).

Information security requirements are developed to ensure a standard and **acceptable** level of confidentiality, integrity, availability, and accountability is maintained over information processed, stored, or communicated by your automated resources.

The definition of requirements that will result in an “acceptable” level of security can only result from a detailed analysis of the criticality of the information resources and threat scenarios. For example, is an intentional attack such as a denial of service or unauthorized access an allowable event? On which servers and under which circumstances? Should such events trigger an alarm? Who should receive it? How will the alarm be communicated? What action should be taken? How much time can elapse before remedial action is taken?

Answers to question like these can provide security implementation guidelines/requirements, but frequently such requirements fail to provide assurance of acceptable security due to the highly dynamic, changing nature of intentional threats. An alternative solution is to base security requirements on “standards” representing “best practices” experienced in the industry.

## ❑ Where to find security standards

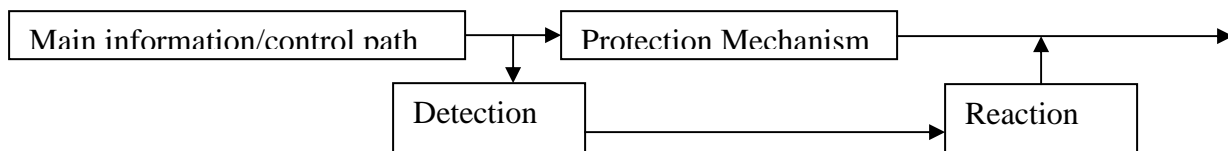
Standards and requirements for automated systems security are generally documented in a comprehensive security policy that is the result of synthesis of the enterprise’s business processes, system asset sensitivity/criticality analysis and the threats posed which risk the confidentiality, integrity and/or availability requirements associated with these assets. Many of these standards are formalized in regulations imposed on specific industries, e.g., HIPAA Security Standards for the healthcare industry or by government agencies.

The scope of these documents will vary from general system requirements to requirements for specific system platforms and networks and call for more comprehensive requirements as the level of criticality of the assets to be protected increases. (See for example the literature available on the [www.sans.org](http://www.sans.org) website for a discussion of levels of security requirements balancing cost of security mechanisms with the risk of loss associated with threats.)

Based on the assessments made during Step 2 - Gather Basic Facts, it may be necessary to offer the information **more** protection than required by your organization's standards referenced above. Standards produced by the requirements usually represent the **minimum requirements** for their particular area. All company systems are required to comply with at least the foundation standards document. If a more specific standard addressing your system is posted (i.e., Minimum Security Configuration Standards for Mid-Range UNIX Systems) then it too (in addition to the minimum requirements) applies.

## ❑ A useful Security Systems Architecture

A useful architecture for the purpose of security requirements analysis is<sup>1</sup>:



Protection mechanisms such as firewalls, password controls, cipher locks and cryptography are designed to deny access. Detection mechanisms such as intrusion detection systems (IDS) look for signs of attack, improper activity or policy non-compliance, and reaction mechanisms, such as automated system reconfigurations to deny access to an attacker, or triggering email (if not on the same system) or phone/pager notifications to appropriate staff, respond via a predefined set of rules for action in the event that a violation of security policy does in fact occur. The allocation of security system **functional** and **performance** requirements to this architecture will provide insight into key security implementation trade-offs balancing cost and risk. For example, it is an accepted information security principal that in today's ever-changing technological landscape, (faster computers, higher bandwidth, new attack tools, etc.), protective mechanisms cannot provide 100% protection. As a result one cannot count on protective mechanisms to assure adequate security of mission critical assets. For the adequate protection of such assets, requirements that assure near-real-time detection and response capabilities may be necessary. For example, one could require that in the event of a detected intrusion, the system launch security assessment scans against other segments of the enterprise to detect other vulnerabilities to that type of attack, and automatically reconfigure the network to block misuse. The ability of a system to automatically detect an event requiring such response may itself be a challenge, frequently requiring the correlation and analysis of data from multiple security devices (e.g., IDSs, firewalls, routers) to accurately detect an attack from a multitude of "false positives".

## ❑ Action required to complete Step 3

Complete the "Information Security Requirements" section of the Worksheet.

## Step 4 - Design / Build System

This Risk Management step takes the security requirements and the allocation to the above-described functional architecture and incorporates technical (hardware/software) and non-technical (physical/procedures) controls into the system's platform and environment to satisfy them. It should be noted that there's a difference between requirements and controls. Requirements are things that the system must do or provide - services. A requirement is usually written in a manner that does not dictate how it will be met. Controls are technical or non-technical mechanisms that satisfy requirements. They are the system developer's way of satisfying the requirements. For example, a requirement might be that "only authorized personnel are allowed access to the system." Controls that satisfy that requirement might be

<sup>1</sup> Winn Schwartau, "Time Based Security", Interpact Press, 1999

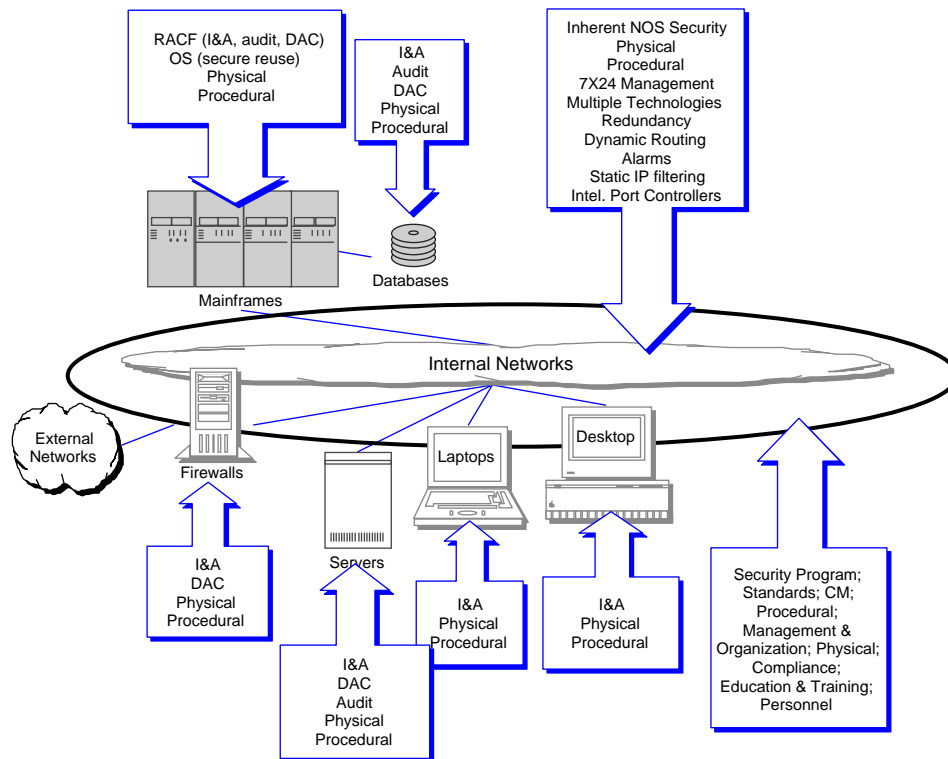
non-technical such as physical controls at entry doors and/or employee badges. Technical controls such as Resource Access Control Facility (RACF) or Enterprise Security Manager (ESM) that require a unique user ID and password before access is granted is another way to satisfy the requirement. Note that when the two types of controls are used together, they form a tighter security perimeter than either of them could by themselves.

Occasionally, requirements will be written in such a way that they drive the implementation solution or the capabilities of the chosen platform will restrict the developer's creativity. More often than not, economics will play a key role in the control options available. In any case, it's up to the developer to orchestrate the available controls to maximize the protection afforded to the system.

### Allocation of Controls

Security controls, both technical and non-technical, should be allocated to system components as the design matures. Control allocations should be economically layered so that there is no single point of failure in the security protection. An ideal layering technique is the integration of technical and non-technical controls so that a potential weakness (vulnerability) in one type of control is offset by the strength in another.

Figure 3 illustrates how different security controls implemented on various system components combine to form an economic and effective security barrier. Note the "system" context where a smaller "system" can, and often is, a part of a larger "system." Maintain a global perspective throughout your risk management endeavors.



**Figure 3. Allocation of Security Controls by Component.**

Examples of control allocations to system components (physical/procedural, processors, LAN/WAN) are provided in Attachment 2 to assist with the selection of security controls available to meet your system security requirements.

### ❑ Action required to complete Step 4

Complete the “**System Description and Architecture**”, “**Implemented Technical Security Controls**”, and “**Implemented Non-Technical Security Controls**” sections of the Worksheet. Use the space provided or attach additional sheets as necessary.

When showing the placement of security controls, don’t forget the non-technical controls. Refer to the illustration above to ensure you are giving yourself maximum credit for implemented security controls.

## Step 5 - Test

Testing against the requirements validates how well controls have been implemented into the system’s design. There are many different ways to “test” whether or not the technical (platform) and non-technical (environment) controls do what they are supposed to do. The following paragraphs discuss different methods of “testing” that can be performed.

### ❑ Methods of Testing

Testing the security implementation of a platform or environment can be accomplished by several methods such as analysis, test, demonstration, inspection, or interview. The methods are listed in order of complexity and depth from most to least. If more than one method exists to test a requirement, the tester should use the more empirical method (or both could be used). For example, satisfying a requirement for annual security training can be verified by either an **interview** of the person responsible for training or an **inspection** of the documentation files to see if it was performed. The inspection would be more thorough and therefore the more appropriate method. Table 3 illustrates testing options that can be used for technical and non-technical controls. The paragraphs that follow describe each of the test methods.

**Table 3. Test Methods.**

Method	Technical Controls		Non-technical Controls	
	Hardware	Software	Physical Environment	Operational and Administrative Procedures
Analysis	X	X		
Test	X	X		
Demonstration *	X	X	X	X
Inspection *		X	X	X
Interview *		X	X	X
* Can be driven and recorded using checklists.				

❑ **Analysis** evaluates the appropriate component using recognized analytical techniques. It compares analytical results with requirements. Testers draw analytical conclusions from test data and extend the analysis of test data to untested conditions. An example of analysis would be the study and examination of software and hardware flow diagrams, design specifications, system audit trails and audit logs.

- ❑ **Testing** involves an evaluation through systematic measurement under all appropriate conditions. Testing security mechanisms typically employs scenario generations designed to generate audit reports, trigger alarms or send messages to security personnel. The granularity of testing is weighed against economic and time constraints.
- ❑ **Demonstrations** are used to observe a predictable event, initiated by a specific input or set of inputs that will always yield the same output or response. Testers may use live or simulated data, actions, or a lack of action to stimulate the predicted response. Demonstrations are used to exercise a control or security feature to verify it can carry out its designed purpose. For example, demonstrate (show me) that I'm locked out of the system after the third incorrect password attempt. Demonstrations can be driven by and recorded using standardized checklists.
- ❑ **Inspection** involves a physical examination of an item, a review of descriptive documentation, and/or a comparison of security characteristics to the corresponding security requirements to verify conformance of the implementation to the requirement. Inspection also includes the verification of accuracy and completeness of documentation or records as well as a mechanical inspection of equipment. Examples of an inspection would be the visual and physical inspection of an output display, entry control points, and cipher locks. Checklists can be used to drive and record inspection-testing methods.
- ❑ **Interviews** provide an effective means of validating the results obtained from other test methods. Interviews include discussions with appropriate personnel to determine adequacy of administrative and procedural requirements. For example, an initial security training requirement may exist which can be verified by inspection of training materials and documentation, but to validate the effectiveness of the training, interviews with users would be necessary. By itself, the interview method in most cases is the least desirable method of testing, but, when combined with another method, will give a better overall "picture" of the security environment. The danger of relying on **only** the interview method of testing is that it leads to a false security state that feeds on the premise that "if we think we're OK then we're OK." Interview questions and answers can be driven and recorded using checklists.

## ❑ Automated Tools for Testing

The use of automated tools to test, and assist in general with risk management, is encouraged. Automated tools provide a level of thoroughness, standardization, and repeatability that is difficult to gain from manual means. Automated tools that measure strength of security controls (such as passwords) or configurations (are compliant with standards and are the correct patches applied) are extremely useful and economical. Automated tools are available to assist with the testing of both technical and non-technical security mechanisms.

In the UNIX world for example, tools such as Computer Oracle and Password System (COPS), Crack, and Tripwire automate testing of technical security mechanisms.<sup>1</sup> COPS and Tripwire incorporate a three-pronged approach in identifying system vulnerabilities: static audits of the operating system search for known loopholes frequently used by hackers, integrity checks of the operating system executable files to ensure they have not been altered, and password file checks to look for weak user passwords and default system or application passwords. Crack is strictly a password cracking routine, which reports on users with guessable or dictionary-like passwords.

Security features and capabilities of security management applications such as RACF, ESM, Intruder Alert (ITA), and Secure Operating Systems (SeOS) can also be used in varying degrees to reinforce the testing of technical controls.

---

<sup>1</sup> Even Security Administrator Tool for Analyzing Networks (SATAN) can be used.

Automated tools for non-technical security mechanisms such as RiskWatch can be used to automate interviews using a standardized question set.

### Action required to complete Step 5

Use checklists produced from the appropriate requirements source with other test methods as necessary to validate that the requirements are met. Complete the “**Testing**” section of the Worksheet and attach supporting testing documentation.

To minimize duplication of effort, consider using subsets of operational tests to satisfy security requirements.

Note: Security testing differs from operational testing. Operational testing requires strict adherence to a set of procedures to specifically test stated requirements or specifications. The goal of operational testing is to determine how well the system meets the operational specifications. Security testing has two perspectives. First, security testing ensures the system meets its security requirements (e.g., locks out an unauthorized user after three unsuccessful login attempts). This is called **presence of mechanism testing**. This second perspective of security testing is called **strength of mechanism testing**. This perspective attempts to circumvent or defeat security mechanisms to identify additional vulnerabilities (e.g., in the case where user identification and authentication (I&A) services are provided by network resources, depressing control + “C” will interrupt a scripted login to network I&A facilities and result in unintended access to the internal hard drive).

Note: Even though the same skills are used for operational testing and security testing, it may be difficult for “operational” testers to take on the “break it” perspective required for strength of mechanism testing. This form of security testing is often similar to “hacking.”

Additional assistance can be obtained from your organization’s security support services personnel.

## Step 6 - Train

No controls can be effective unless users and administrators are trained in their use. The lack of security awareness, training, and education is one of the very few controls that can be successfully argued as both a threat and a vulnerability. Training programs ensure that administrators and users have the knowledge they need to do their job and provide a basis upon which remedial or disciplinary action can be taken if necessary.

Security awareness, training, and education programs seek to provide appropriate levels of training to all positions. Table 4 illustrates a sample of various levels of proficiency to ensure training is incorporated into your security strategy as an effective control.

### Action required to complete Step 6

Ensure that effective security awareness, training, and education is conducted. Complete the “**Training**” section of the Worksheet.

**Assistance can be obtained from your organization’s security awareness, training, and education personnel.**

## Step 7 - Assess

The desired outcome of this step is an informed self-assessment of the risks associated with operating the system and whether or not those risks are acceptable. The assessment is supported by the previous

process steps; determining the sensitivity of the information, determining the criticality of the system to your company, knowing what the system is supposed to do to protect the information, testing the controls to see how well they work, and training administration and using personnel on their duties. You might call this step your “security report card.” The assessment step is where the “rubber hits the road” in risk management. This is where all of the previous work comes together; the diamond in the Risk Management Model.

During this step you will be performing a self-assessment of not only the individual controls but also an overall acceptability assessment of the risks of operating the system.

### ❑ Effectiveness Table

Table 5 provides values to quantify the effectiveness of individual controls, both technical and non-technical. Although the descriptions are general in nature, attaching a numeric value to each of the controls provides a degree of data reduction that will help you arrive at a final assessment of overall system risk.

**Table 4. Sample Level of Training Requirements.**

		Training Content Areas				
		Computer Security Basics	Security Planning & Management	Computer Security Policy & Procedures	Contingency Planning	Systems Life Cycle Management
Audience						
Top Executives						
Program & Senior Managers						
Security, Systems Integrity & Audit Personnel						
Company Management & Operations Personnel						
Company Employees and In-house Contractors						

Key to Training Levels	
	Awareness - Sensitivity to the system's threats, vulnerabilities, and the need for system security within your company.
	Policy - Understanding computer security principles as an aid for decision-making.
	Implementation - Ability to recognize and assess the threats and vulnerabilities to the system's resources to set security requirements.
	Performance - Skill required to design, execute, or evaluate system's security procedures and practices.

(Adopted From National Institute of Standards and Technology Special Publication 500-172)

**Table 5. Values for Assessing Control Effectiveness.**

Effectiveness Value:	Description:
5	Controls are verified to be fully implemented and to be effective. (They satisfy the requirement.) Controls are compliant with company Standards and Practices.
4	Controls are present, appear to be effective, but are not verified by testing.
3	Controls are implemented but are only partially effective.
2	Controls are not fully implemented. Full control functionality is missing on all required components. Where implemented, controls are only partially effective.
1	Controls are only partly implemented and are not effective in satisfying their requirements.
0	The controls are not implemented.

### ❑ Action required to complete Step 7

Note that there are two parts to the “**Assessment**” section of the Risk Management Worksheet (in addition to results of previous assessments). First an assessment of individual control effectiveness and second an overall acceptability assessment of system risk.

Using the testing results of **Step 5 - Test**, assign an effectiveness value for each of the system’s technical and non-technical controls identified in **Step 4 - Design / Build System** using the table above. Place the assessed value in the appropriate technical or non-technical control effectiveness assessment portions of the “**Assessment**” section of the Worksheet.

Determine an overall system risk assessment. Compare, weigh, and analyze individual control assessments with the sensitivity, criticality, threats, and other facts identified in **Step 2 - Gather Basic Facts** and the requirements identified in **Step 3 - Establish Requirements**. For example, a low effectiveness of an identification and authentication (I&A) control might provide an avenue for exploitation by a human-intentional threat and the compromise of sensitive information.

Finish the “**Assessment**” section of the Worksheet. The system owner will sign the Owner’s Risk Assessment Statement.

Assistance for this step can be obtained from your organization’s risk management personnel.

## Step 8 - Certify

Certification is an important part of the Risk Management Process. It indicates to others that controls work as advertised. Also important is the signature of the system owner/developer that establishes individual accountability.

It is important to note that while the certification signature indicates that controls work as advertised, it does not guarantee flawless security or impenetrability. What it does indicate is that the description of the control effectiveness on the Worksheet is representative of the way things really are.

### **Action required to complete Step 8**

The system's owner/developer will sign the "**Certification**" section of the Worksheet

## **Step 9 - Monitor and Maintain**

The final step ensures the security posture of the system remains at an acceptable level throughout its life cycle. This step requires continued surveillance of the Risk Management Model components to ensure that changes in the system or threats do not decrease its ability to protect the information. Changes to watch out for include application and operating system updates, changes in user interfaces, and changes in the nature of the information.

Periodically, vendors and/or computer emergency response teams will publish, to the general public, system vulnerabilities that have been reported to them. Along with a high-level description of the reported vulnerability is a "patch" to control it. Maintaining and applying the current patches is part of this step.

Recurring administrator and user training is equally important to keeping the security controls up to date.

### **Action required to complete Step 9**

Periodically (at least annually is recommended) review and update as necessary the Risk Management Worksheet and ensure that the level of protection afforded the information remains at an acceptable level throughout the system's life cycle.



# Attachment 1 - Risk Management Worksheet

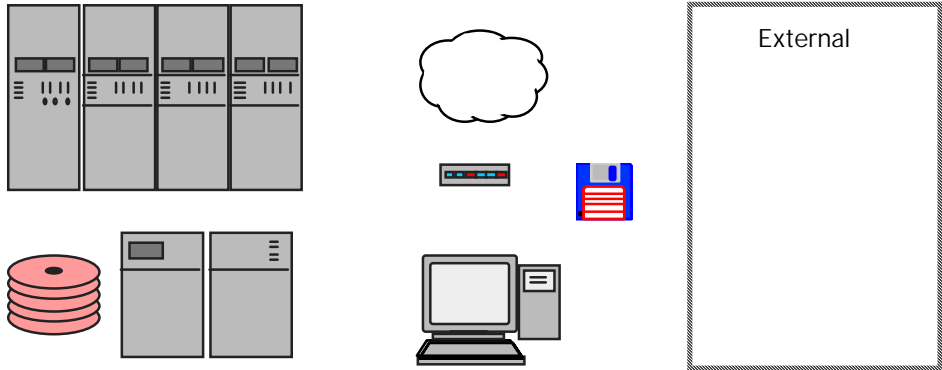
<b>1. General Information</b>	
System name:	
System IP and node name:	
System category:	<input type="checkbox"/> Office Automation (COTS products, local storage, and network) <input type="checkbox"/> Major Application (Specialized custom / semi-custom software) <input type="checkbox"/> General Support System (Platform only with utilities, database) <input type="checkbox"/> Specialized System (Switch, router, network server, etc.) <input type="checkbox"/> Other
System status:	<input type="checkbox"/> Operational <input type="checkbox"/> Under development <input type="checkbox"/> Undergoing major modification <input type="checkbox"/> Other
General purpose of the system:	
System owner/ Organization:	
System developer / organization:	
System operator / organization:	
Person with overall responsibility for system security:	
Physical location of the system (e.g., data center):	



<b>2a. Sensitivity and Criticality Information</b>	
Data sensitivity:	<input type="checkbox"/> No sensitivity level <input type="checkbox"/> Organization Internal Use Only <input type="checkbox"/> Organization Confidential <input type="checkbox"/> Organization Restricted <input type="checkbox"/> Organization Personal and Confidential <input type="checkbox"/> Other _____ (customer-driven)
Business impact if system fails:	<input type="checkbox"/> Number of customers affected: _____ <input type="checkbox"/> Revenue / month lost: _____ <input type="checkbox"/> Calls / month lost: _____ <input type="checkbox"/> Other: _____
System criticality:	<input type="checkbox"/> 0 Negligible impact on company revenue, reputation, or operations. <input type="checkbox"/> 1 Minimal impact on company revenue, reputation, or operations. <input type="checkbox"/> 2 Adverse impacts on company revenue, reputation, or operations. <input type="checkbox"/> 3 Irreparable impacts on company revenue, reputation, or operations.
<b>2b. Threat Information</b>	
Greatest area of threat concern:	<input type="checkbox"/> Human-Unintentional (H-U) (data entry and procedural errors) <input type="checkbox"/> Human-Intentional (H-I) (hackers, industry spies, malicious) <input type="checkbox"/> Structural (S) (configuration, integration, power problems) <input type="checkbox"/> Natural (N) (fire, flood, wind, seismic)
Specific system threats by type:	<input type="checkbox"/> H-U _____ <input type="checkbox"/> H-I _____ <input type="checkbox"/> S _____ <input type="checkbox"/> N _____
Threat basis:	<input type="checkbox"/> Based on general knowledge. <input type="checkbox"/> Based on actual instances (Explain)



<b>2c. User Community Information</b>	
Do others share the system's information?	<input type="checkbox"/> Yes <input type="checkbox"/> No
Who are they?	<input type="checkbox"/> Internal organization _____ <input type="checkbox"/> Organization/Contractor/Vendor _____ <input type="checkbox"/> Organization/Alliances _____ <input type="checkbox"/> General Public _____ <input type="checkbox"/> Other _____
Current interfaces to other company systems:	
Planned interfaces to other systems:	
<b>3. Information Security Requirements</b>	
What standards have been used for the system's information security requirements, design and / or operation?	<input type="checkbox"/> Minimum Security Baseline Standard for Information Systems <input type="checkbox"/> Information Asset Security Standards <input type="checkbox"/> Network and Systems Security Requirements and Standards <input type="checkbox"/> Local Area Network Security Standards <input type="checkbox"/> Internet Gateway Security Standards <input type="checkbox"/> Company Security Evaluation Process <input type="checkbox"/> Standards for Application Security <input type="checkbox"/> Minimum Security Config. Standards for Mid-Range VMS <input type="checkbox"/> Minimum Security Config. Standards for Mid-Range UNIX <input type="checkbox"/> Other _____ <input type="checkbox"/> Other _____ <input type="checkbox"/> Other _____
Have exceptions been requested / approved?	<input type="checkbox"/> Not requested <input type="checkbox"/> Requested (explain) <input type="checkbox"/> Requested and approved

4a. System Description and Architecture				
		Model	O/S	S/W
Primary platform components:	<input type="checkbox"/> Mainframe	_____	_____	_____
	<input type="checkbox"/> Midrange	_____	_____	_____
	<input type="checkbox"/> PC/Workstation	_____	_____	_____
	<input type="checkbox"/> Switch	_____	_____	_____
	<input type="checkbox"/> Gateway/router	_____	_____	_____
	<input type="checkbox"/> Server	_____	_____	_____
	<input type="checkbox"/> Database	_____	_____	_____
	<input type="checkbox"/> Network	_____	_____	_____
	<input type="checkbox"/> Other	_____	_____	_____
Diagram of basic physical architecture and placement of security controls:				
	<input type="checkbox"/> Other			
Software:	<input type="checkbox"/> Major Applications	_____	_____	Security Controls? _____
		_____	_____	_____
		_____	_____	_____
	<input type="checkbox"/> Support Applications	_____	_____	_____
		_____	_____	_____



Communication interfaces:	<input type="checkbox"/> Internal (local LAN) <input type="checkbox"/> Company-wide (Infolink) <input type="checkbox"/> Production Network <input type="checkbox"/> Unsecure dial-up <input type="checkbox"/> Secure dial-up <input type="checkbox"/> Internet <input type="checkbox"/> Other
<b>4b. Implemented Technical Security Controls</b>	
Are there existing technical security requirements?	<input type="checkbox"/> No requirements <input type="checkbox"/> Organization minimum requirements <input type="checkbox"/> Organization minimums + (internal) <input type="checkbox"/> Organization minimums + (customer-driven) <input type="checkbox"/> Other
Are the requirements documented?	<input type="checkbox"/> Yes <input type="checkbox"/> No
Are there I&A controls?	<input type="checkbox"/> Yes <input type="checkbox"/> No
At what point is the user prompted for a userID and password?	<input type="checkbox"/> Never <input type="checkbox"/> Signing on to the PC/workstation <input type="checkbox"/> Signing on to the network <input type="checkbox"/> Signing on to an enroute server <input type="checkbox"/> Signing on to the mid-range/mainframe <input type="checkbox"/> Signing into the database <input type="checkbox"/> Other
Are there times when only a userID is requested?	<input type="checkbox"/> Yes <input type="checkbox"/> No
Are multiple user accounts in use?	<input type="checkbox"/> Yes <input type="checkbox"/> No
Are there discretionary access controls?	<input type="checkbox"/> Yes <input type="checkbox"/> No
Are audit trails generated?	<input type="checkbox"/> Yes <input type="checkbox"/> No
Is audit reduction support available?	<input type="checkbox"/> Yes <input type="checkbox"/> No
Can / does the system provide an end-to-end audit of a single user?	<input type="checkbox"/> Yes <input type="checkbox"/> No

Can / does the platform perform secured object reuse?	<input type="checkbox"/> Yes <input type="checkbox"/> No
Is secured object reuse turned on?	<input type="checkbox"/> Yes <input type="checkbox"/> No
Are system back-ups performed?	<input type="checkbox"/> Yes <input type="checkbox"/> No
When are back-ups performed?	
Where are they kept?	
Does the platform have encryption controls?	<input type="checkbox"/> Yes <input type="checkbox"/> No
Describe encryption performed:	
Does the platform have virus detection and eradication software?	<input type="checkbox"/> Yes (Name product/version _____) <input type="checkbox"/> No
<b>4c. Implemented Non-Technical Security Controls</b>	
Have non-technical security requirements been levied on the system?	<input type="checkbox"/> No requirements <input type="checkbox"/> Organization minimum requirements <input type="checkbox"/> Organization minimums + (internal) <input type="checkbox"/> Organization minimums + (customer-driven) <input type="checkbox"/> Other
Have system security duties and responsibilities been assigned?	<input type="checkbox"/> Yes <input type="checkbox"/> No
Have physical security controls been implemented?	<input type="checkbox"/> Yes (Describe: _____) <input type="checkbox"/> No
Are instructions for performing security functions documented?	<input type="checkbox"/> Yes (Describe: _____) <input type="checkbox"/> No
Are audit trail reviews performed on a	<input type="checkbox"/> Yes (When: _____) <input type="checkbox"/> No

routine basis?	
Is there an incident response capability?	<input type="checkbox"/> Yes <input type="checkbox"/> No
Who receives the AIS security incident report?	
Are contingency plans documented?	<input type="checkbox"/> Yes <input type="checkbox"/> No
<b>5. Testing</b>	
Have the system's security controls been tested?	<input type="checkbox"/> Yes (Attach results) <input type="checkbox"/> No
Who performed the tests?	
What methods of "testing" were used?	<input type="checkbox"/> Analysis <input type="checkbox"/> Test <input type="checkbox"/> Demonstration <input type="checkbox"/> Inspection <input type="checkbox"/> Interview <input type="checkbox"/> Other: _____
Testing was conducted using:	<input type="checkbox"/> Security-specific checklists <input type="checkbox"/> Operational test scenarios written at the keystroke level <input type="checkbox"/> Security test scenarios written at the keystroke level <input type="checkbox"/> Penetration / hacking techniques <input type="checkbox"/> Other: _____
Were automated security tools used?	<input type="checkbox"/> Yes (Name/Version: _____) <input type="checkbox"/> No
<b>6. Training</b>	
Are security administrators trained?	<input type="checkbox"/> Yes <input type="checkbox"/> No
Do users receive security training?	<input type="checkbox"/> Yes <input type="checkbox"/> No
Is recurring security training provided on a regular basis?	<input type="checkbox"/> Yes <input type="checkbox"/> No
<b>7. Assessment</b>	
Previous risk assessment?	<input type="checkbox"/> Yes (when) <input type="checkbox"/> No



Risk Management Questions

	<input type="checkbox"/> Other
Outcome of previous risk assessment:	<input type="checkbox"/> N/A <input type="checkbox"/> Acceptable risk <input type="checkbox"/> Unacceptable risk (attach description of corrective action)
Effectiveness assessment of technical controls:	<input type="checkbox"/> Identification & Authentication _____ <input type="checkbox"/> Audit Capture _____ <input type="checkbox"/> Discretionary Access Control _____ <input type="checkbox"/> Encryption _____ <input type="checkbox"/> Back-up _____ <input type="checkbox"/> Secure Object Reuse _____ <input type="checkbox"/> Security Software _____ <input type="checkbox"/> Other _____ <input type="checkbox"/> Other _____ <input type="checkbox"/> Other _____
Effectiveness assessment of non-technical controls:	<input type="checkbox"/> Configuration Management _____ <input type="checkbox"/> Security Management & Administration _____ <input type="checkbox"/> Standards and Practices _____ <input type="checkbox"/> Awareness, Training, and Education _____ <input type="checkbox"/> Procedural Security _____ <input type="checkbox"/> Physical Security _____ <input type="checkbox"/> Other _____ <input type="checkbox"/> Other _____ <input type="checkbox"/> Other _____ <input type="checkbox"/> Other _____
Owner's Risk Assessment Statement:	<input type="checkbox"/> I find the overall system risks to be Acceptable <input type="checkbox"/> I find the overall system risks to be Unacceptable (Attach rationale and proposed corrective actions.)  _____



<b>8. Certification</b>	
Owner's / Developer's Certification:	<p>I certify that the security controls in the above system have been implemented and work as stated; and that they provide a level of protection for information and system assets consistent with the intent of the company's Information Security Policy.</p> <p>_____</p>
<b>9. Monitor and Maintain</b>	
Is there a periodic review process for the system's security?	<input type="checkbox"/> Yes <input type="checkbox"/> No
<b>Management Use:</b>	
Concur with assessment:	<input type="checkbox"/> Yes <input type="checkbox"/> No
Next scheduled review:	
Remarks:	



## Attachment 2 - Control Allocation

### Control Allocation

Security controls are allocated during design to various architectural components of the system. Some controls are allocated to the platform (technical controls) and some to the environment (non-technical controls). Controls vary in the services they provide. Some services are visible to the users while others are not. The following table provides examples of technical and non-technical security controls, where they might be applied within a system’s architecture, and the types of services they provide.

Architecture Component	Security Services Required	Security Controls Allocated
Physical & Procedural <ul style="list-style-type: none"> <li>• Physical Facility</li> <li>• Facility Support (fire, and so forth)</li> <li>• Organizational Personnel</li> <li>• Organizational Procedures</li> </ul>	Identification and Authentication	Picture Badges Personal recognition
	Access Control	Guarded entry Door keys and cipher locks Surveillance by coworkers and security administrators Hardware locks Online warnings of access rules
	Confidentiality	Trash storage and removal Controlling video displays and printing devices (ribbons, copy counting, labeling)
	Integrity	Physical inspections of equipment Protection of software masters and small components Configuration management
	Availability	Physical inspections of equipment Protection of software masters Contingency plan testing Backups Ensure virus signatures and OS patches are up-to-date Incidence response team and response procedures



<ul style="list-style-type: none"> <li>• Workstations</li> <li>• PCs</li> <li>• Mid-range</li> <li>• Mainframes</li> <li>• Application Servers</li> <li>• Operating Systems</li> <li>• Database Management Systems</li> </ul>	Identification and Authentication	User ID and Passwords Security tokens Biometrics Security Software Automatic logoff Digital Certificates
	Access Control	Encryption User ID and Password enforcement Security tokens (role/user-based access) Separation of duties (to minimize fraud) Defined user shells Defined user permissions Discretionary Access Control (DAC) Access control lists Warning banner Audit System users, events and open ports Run password-cracking software
	Confidentiality	Security Software Discretionary Access Controls Object Reuse Audit Records
	Integrity	System diagnostics Non-forgable seals on cases Software checksums/CRCs Audit Records Digital Signatures
	Availability	Processor redundancy Diversity Backups for contingency operations Filters to detect and eradicate viruses Enable logging of system events
<ul style="list-style-type: none"> <li>• Multiplexors</li> <li>• Routers</li> <li>• Switches</li> <li>• Message Transfer Agents</li> <li>• Network Operating Systems</li> </ul>	Identification and Authentication	User ID and Passwords Security token technology
	Access Control	User ID and Passwords Firewalls Warning banner Separation of duties Router Access Control Lists Network Scans
	Confidentiality	Discretionary Access Controls Encryption/VPN
	Integrity	Configuration management System self-diagnostics Non-forgable seals on cases Software checksums/digital signatures
	Availability	Redundancy Diversity Backups for contingency operations

Communications Networks	Identification and Authentication	User ID and Passwords Secured entry points
<ul style="list-style-type: none"> <li>• Transmission Systems</li> </ul>	Access Control	User ID and Passwords Audit records Firewalls Network Scans Intrusion Detection
<ul style="list-style-type: none"> <li>• Network-specific Servers</li> </ul>	Confidentiality	Encryption/VPN Secure modem entry
<ul style="list-style-type: none"> <li>• Network Mgt Systems</li> </ul>	Integrity	Network management Digital Signatures
<ul style="list-style-type: none"> <li>• Switches</li> <li>• Routers</li> <li>• Gateways</li> </ul>	Availability	Redundancy Diversity Backups for contingency operations Router Access Control Lists Intrusion Detection