

Managing Information Technology Security Risk

David Gilliam

Jet Propulsion Laboratory, California Institute of Technology

david.p.gilliam@jpl.nasa.gov

Abstract. Information Technology (IT) Security Risk Management is a critical task for the organization to protect against the loss of confidentiality, integrity, and availability of IT resources. As systems become more complex and diverse and attacks from intrusions and malicious content increase, it is becoming increasingly difficult to manage IT security risk. This paper describes a two-pronged approach in addressing IT security risk and risk management in the organization: 1) an institutional enterprise approach, and 2) a project life cycle approach. The institutional approach addresses automating the process of providing and maintaining security for IT systems and the data they contain. The project life cycle approach addresses providing semi-automated means for integrating security into the project life cycle. It describes use of a risk tool, the Defect Detection and Prevention (DDP) tool developed at the Jet Propulsion Laboratory, to manage risks. It also describes project and institutional mitigation processes and tools.

1. Introduction

Engineering Information Technology (IT) security is a critical task to manage in the organization. With the growing number of system security defects being discovered and as the impact of malicious code escalates, Security Engineering (SE) of IT security is increasingly critical both organizationally and in the project life cycle [1]. Organizations have suffered significantly over the last few years due to the loss of Confidentiality, Integrity, and Availability (CIA) of IT resources due to malicious code attacks and breakins. Understanding and mitigating these risks is paramount in protecting organizational resources. The problem has been noted by the United States (US) Government Accounting Office (GAO) showing that US federal agencies are at high risk. In a recent audit of the US Department of Defense (DoD), the GAO reported, "Security assessments continue to identify weaknesses that could seriously jeopardize DoD's operations and compromise the confidentiality, integrity, or availability (CIA) of sensitive information. . . . Specifically, the Inspector General found security lapses relating to access to data, risk assessments, sensitive data identification, access controls, password management, audit logs, application development and change controls, segregation of duties, service continuity, and system software controls, among others." [2] A single compromise can put an entire organization at risk of loss of IT resources whether it is from a system breakin, a worm infection, or an unintended exposure—the 'weakest link' syndrome.

Understanding and managing IT security risk becomes paramount in protecting organizational resources—systems, data, facilities, and most of all people. Controls to identify and manage IT security risks are available, but they are generally applied non-uniformly as a reaction to some threat. Few organizations perform a risk assessment and analysis along with a review and implementation of mitigations and their effectiveness to those threats. As systems grow more complex and distributed, managing the IT environment and its resources securely is increasingly problematic. A concerted approach to managing IT security risk is needed to understand the extent of risks, available mitigations, and their relative value versus costs. There are two primary areas that IT security risk management needs to address: 1) management of risk for systems and the data on them at an organizational level; 2) management of risk in the System Development Life Cycle (SDLC) in producing goods and services. Anything that is produced and consumed needs to have security 'built-in' as opposed to being 'bolted-on.' Risk management has an impact on corporate survival to some degree.

SE is a continuous life cycle process that extends from the organization to the project. Risks at the institutional level impact projects and risks at the project level impact the institution. Both areas must be managed through an SE approach. All too often approaches to managing IT security either address the enterprise environment or address the project life cycle singularly. What may be appropriate for the enterprise (implementation of firewalls, patch management, etc.) may not directly relate to the project life

cycle, especially the system development life cycle (SDLC) which is more concerned with implementation of security controls and reducing vulnerabilities and exposures in systems and software. [3] If a company practices 'dog fooding' (using their own systems and software), it is vitally important that security risk management and controls be applied as a cooperative effort between these two key areas.

This paper describes an approach for managing IT security risk. It identifies tools and instruments that can be used in the project life cycle to identify and reduce or mitigate IT security risks. Section 2 discusses risk management and a model for calculating risk and risk mitigations. Section 3 describes a Defect Detection and Prevention (DDP) risk assessment and analysis tool used in the project life cycle and the capability for extending it to IT security risk assessment. Section 4 describes the specific application of the DDP tool to IT security risk in the project life cycle. Section 5 describes the use of the tool enterprise-wide. Section 6 concludes with a summation of the need for integration of both project and institutional risk processes. Effective risk management should instantiate a coordinated approach that addresses both areas, and identify the interactions and impact of risks and their mitigations between them.

2. IT Security Risk and Risk Management

An SE approach to IT security will facilitate managing it effectively. Understanding IT security risk and its impact in terms of damage to the organization will help to identify the level of risk that the organization must manage or accept. First, the System Security Engineer (SSE) must understand the nature of risks in terms of vulnerabilities and exposures and their likelihood of being used against the organization along with their impact. Second, the SSE must have a good grasp of mitigation factors, the extent to which risks are mitigated by various technologies and their relative costs. Third, the SSE must be able to provide critical reports to management to obtain needed resources to implement the mitigations and maintain a level of currency in risk management.

2.1. Security Risk

Security risk impacts organizational IT resources. The impact extends to the SDLC in the production of goods and services for consumption either by a client/customer or by the organization. System and software defects can be reduced significantly through risk management. When risk and risk management are used in reference to IT security, the discussions generally focus on defining and describing security risk and mitigations in terms of protection of data and systems. IT security risk management is characterized in terms of Confidentiality, Integrity and Availability (CIA). These terms are commonly defined as:

- *Confidentiality*: Assuring information will be kept secret, with access limited to appropriate persons. For Intellectual property or medical information, confidentiality is a critical issue.
- *Integrity*: Assuring information will not be accidentally or maliciously altered or destroyed. Loss of Integrity is a critical issue for data on which decisions are based.
- *Availability*: Assuring information and communication services will be ready for use when expected. An attack can impact a critical system that is dependent on high availability.

Security risk is similar to other identified key risk areas like safety and reliability. There are numerous definitions of risk. The National Aeronautics and Space Administration (NASA) Continuous Risk Management (CRM) web site defines risk as "characterized by the combination of the probability that a project or other enterprise will experience an undesired event with unacceptable consequences, impact, or severity." Starting with this definition, risk management is defined as "a proactive, continuous and iterative process to manage risk to achieve the planned objectives. The process involves identifying, analyzing, planning, tracking, controlling, documenting, and communicating risks effectively." [4] It is illustrated by CRM as a continuous process as shown in Figure 1.

A risk assessment methodology is needed to aid in and guide this process. The National Institute of Standards and Technology (NIST) "Risk Management Guide for Information Technology Systems," presents a nine step process to risk assessment: 1) System Characterization, 2) Threat Identification, 3) Vulnerability Identification, 4) Control Analysis, 5) Likelihood Determination, 6) Impact Analysis, 7) Risk Determination, 8) Control Recommendations, 9) Results Documentation, with each step having specified inputs and outputs that lead to the succeeding step. [4]

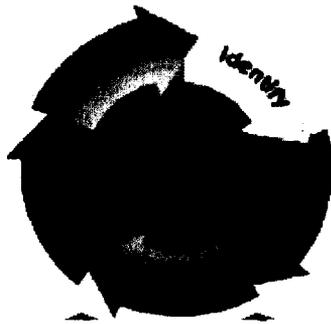


Fig. 1: Risk Assessment Process

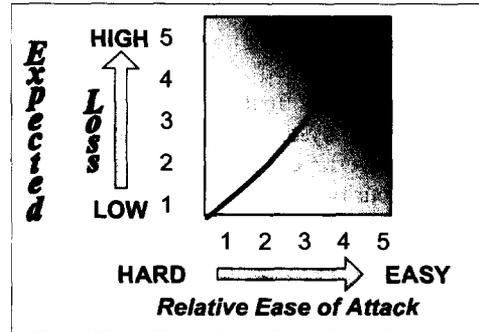


Fig. 2: Exploit Attack Probability

A risk assessment methodology must quantify the cost of a risk occurrence (loss of CIA), and the cost to mitigate risks. One methodology is Probability Risk Assessment (PRA). PRA quantifies risk as: [5]

$$\frac{RISK}{\frac{Detriment}{Unit Time}} = \frac{FREQUENCY}{\frac{Events}{Unit Time}} \times \frac{SEVERITY}{\frac{Detriment}{Event}} \quad (1)$$

In application to security, risk is a function of the *impact* an adverse event would have were it to succeed in breaching defenses, its *likelihood of succeeding*, and the *frequency* at which such events are perpetrated. Quantifying risk in these terms depends on the relative value of the potential loss or disruption should the risk event occur. A formula to quantify IT security risk is defined here as:

$Risk = impact * likelihood * frequency$, where:

$Impact = damage * recovery time$

- Damage can be characterized as the criticality of the data and IT resources along with the degree and extent of their destruction or loss – that is, the criticality of the data and resources and the degree and extent of the loss or compromise. Degree is the damage to a system or set of resources, with extent being the number of systems affected and/or amount of data compromised. A key approach to decreasing risk is to adopt measures that reduce the damage should security attacks succeed in breaching defenses.

- Recovery time is the length of time to recover needed data and IT resources from a compromise

$Likelihood = potential success of an attack$

- Likelihood is the potential that the attack succeeds, and therefore leads to loss or compromise of CIA. A key approach to decreasing risk is to adopt defenses that make attacks less likely to succeed (e.g., training users on selection of passwords so that it is less likely that password hacking will succeed in locating a valid password or applying security patches).

$Frequency = number / time$, where $number = ease * likelihood * impact$

- Number is the number of events occurring over a time interval
- The frequency of an exploit being perpetrated is based on three factors: how easy it is to originate an attack, how likely that attack is to succeed, and how much impact it will have if it does succeed – this combination reflects the malicious intent of would-be attackers.

A consequence of this equation is that *likelihood* and *impact* factors occur twice in the overall formula:

$$Risk = impact * likelihood * frequency = impact * likelihood * (ease * likelihood * impact) = impact^2 * likelihood^2 * ease \quad (2)$$

The key characteristic of SE (compared to safety engineering) is the malicious intent of the attackers, who deliberately favor attacks that they perceive have a greater potential for success and a greater propensity for impact (see Figure 2). Attack sophistication and complexity are unpredictable and these must factor into risks and their mitigations. Damage is premised on the fact that attacks that are easier to carry out and that result in greater harm will occur more often. However, it is difficult to predict new

attacks and attack types. System complexity factors and sophistication of attacks create events that must be evaluated as they occur. For this reason IT security risk management must be a persistent process.

2.2. Security Risk Management

Effective IT security risk management requires collaboration between organizational management, IT security professionals, system engineers, and other stakeholders. It requires knowing and understanding customer needs, government regulations, stakeholder requirements, the organizational environment, etc. Identifying IT security risks, providing a means to manage, and mitigating or accept risks require significant resources and security engineering. It is not the responsibility of just one or two people. The SSE must have the support and involvement of management, system engineers, system administrators, contracts and procurement officers, legal affairs, even the general users.

It is requisite in security engineering to decompose governing policies, standards, and requirements from the customer and stakeholders into their basic constituent elements to assess risks and develop a risk management plan. However, as shown above, the complexity of the IT environment and the sophistication of attacks are impacting the ability to manage these risks. Use of a risk assessment tool like DDP will aid the security engineer in identifying and managing these risks. Additionally, a risk tool that provides a graphical representation of the cost of risks versus the cost of mitigating risks will enable informed decisions on risks that are critical and must be mitigated, those that should be mitigated depending on the available resources, and those that can be accepted as residual risks.

2.3. Related Work

The Gartner Group in identifying the cornerstones of an InfoSec (Information Security) risk management program makes the point that "IT assets that put an enterprise at risk must be identified through an IT risk assessment inventory that covers multiple domains in an organization." [6] Not directly included in their assessment is IT SSE in the SDLC. Other security risk management approaches also address enterprise security risk management from a system or site qualification perspective. [7, 8, 9] Both ISO9000 and the Capability Maturity Model Integration (CMMI) address the importance of managing risk. The CMMI web site provides models for improvement and management of the development life cycle. [10, 11] The Carnegie Mellon University (CMU), Software Engineering Institute (SEI), provides several publications and a method for security risk management called "Octave." [12] The method provides detailed processes, worksheets and guides for an analysis team to conduct a risk evaluation for their organization.

Recently, the National Institute of Health's (NIH) Center for Information Technology (CIT) has taken the problem of IT risk management to another level by providing an application/system security plan template that identifies several types of security controls. [13] The template provides guidance to security personnel, project managers, and system engineers in the steps to integrate security into the institutional processes and the project life cycle and can be used along with the security risk template identified here.

Security engineering is now just beginning to be addressed in the SDLC as depicted by the number of works on the subject being published. [1, 3, 14] These works present a system life cycle approach that addresses requirements, design, development, operations and maintenance. However, many approaches do not cover the relationship and integration of the SDLC and institutional risk management processes. Additionally, the process of phasing out software and systems often is not addressed. When they are phased out, security exposures and vulnerabilities may be introduced, especially if the other systems are dependent on receiving data from them and the people responsible for these systems have not been notified.

3. Defect Detection and Prevention (DDP) Risk Management Tool

A risk management approach at the Jet Propulsion Laboratory (JPL) developed by Steve Cornford and Martin Feather is the Defect Detection and Prevention (DDP) tool. [15, 16] The Defect Detection and Prevention tool is a risk assessment instrument that is analogous to a blank spreadsheet. Inputs into this tool are templates. The goal of DDP is to "facilitate risk management over the entire project life cycle beginning with architectural and advanced technology decisions all the way through operation." [16] "The name reflects its origins as a structured method for planning the quality assurance of hardware systems.

Since then its scope has expanded to also encompass decision-making earlier in the development lifecycle, and to be applicable to software, hardware and systems.” [15] DDP has proven to be effective as a risk management tool as shown by the results of its use on a JPL flight project subsystem. [17] Early project life cycle risk management and mitigation is the core of DDP. Whereas most risk and cost models take time to develop and provide results over the project life cycle, DDP begins early in the life cycle and attempts to provide risk costs and tradeoffs early when initial design decisions are being made.

3.1. Process for Identifying Risks, Mitigations and Relative Weighting of Each

DDP assists system engineers in identifying risks, the relative cost of mitigating the risks and the trade-offs in risk mitigation and acceptance. “DDP explicitly represents risks, the objectives that risks threaten, and the mitigations available for risk reduction. By linking these three concepts, DDP is able to represent and reason about the cost-effectiveness of risk reduction alternatives.” [17] The DDP process brings together stakeholders in the project who are domain experts and who represent the life cycle phases from inception to retirement. According to Feather, “The single most important aspect of the DDP approach is that it supports multiple experts [who] pool their knowledge” allowing them “to take the sum total of their pooled knowledge into account as they make decisions.” [15] In addition, it allows users and domain experts to assign relative weights to risks and risk mitigations. This process represents a multi-disciplinary approach to risk management in the project life cycle—one of the strengths of the DDP process.

The application of DDP to security as a risk management tool will allow an SE more effectively to assess and manage risks whether it is for the institution or for the SDLC. Both institutional and project risks and their mitigations need to be evaluated together for a full risk impact/mitigation assessment. The security engineer needs to work closely with the system engineer and domain experts in this process. Figure 3 depicts the process that facilitates providing weighted inputs into the DDP tool.

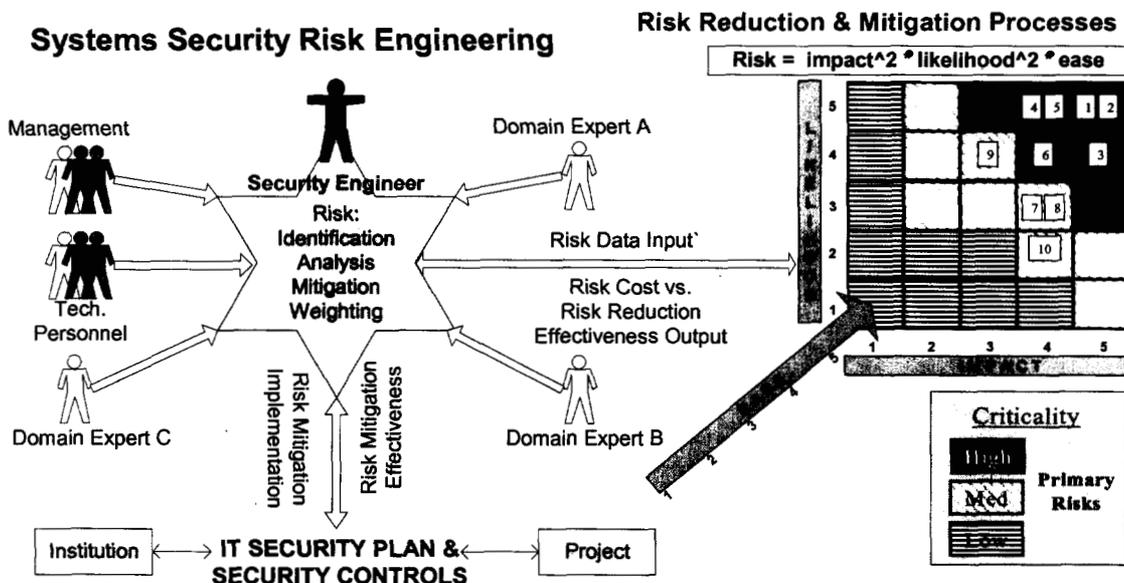


Fig. 3: Security Engineering Risk Assessment/Management Process

The process pools the combined inputs of the domain experts and performs calculations over the entire body of gathered information providing aggregate risk calculation information and searches for near-optimal solutions for mitigating risks. It then provides coherent visualizations back to the domain experts so well-informed decisions can be made. The process can be refined until an optimal solution is achieved. [15] Figure 4 is a graphical representation of the iterative process that identifies the optimal area for costing risks.

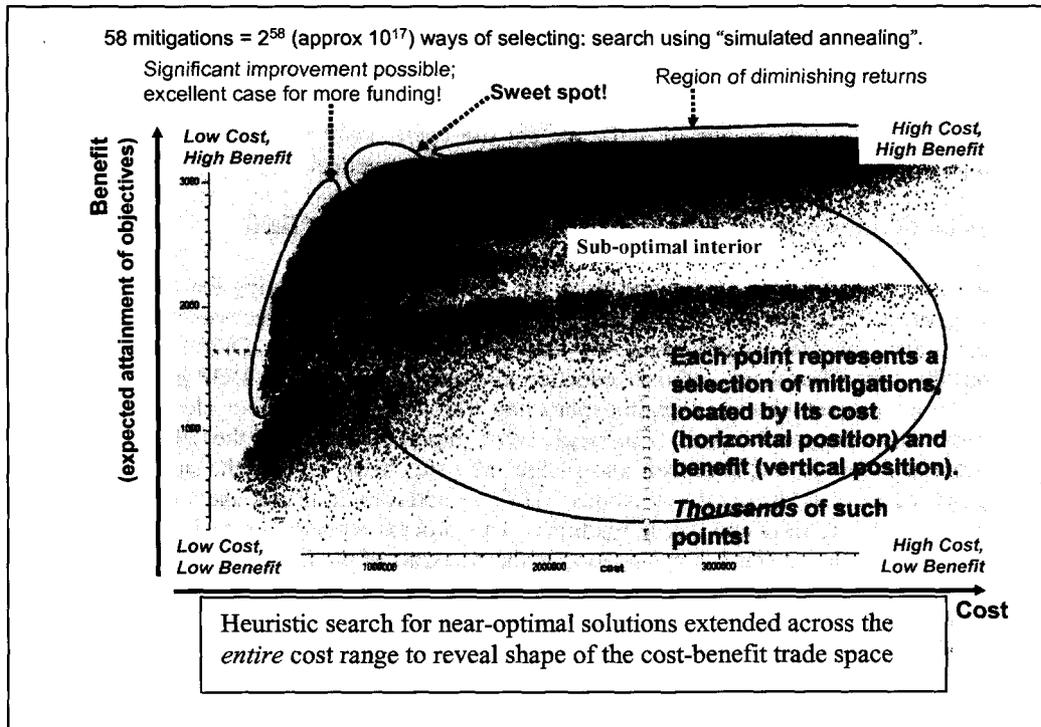


Fig. 4: Cost/Benefit trade-off analysis

The results of a DDP risk assessment/mitigation analysis are output in a graphical representation of the mitigated risks and the residual risks as shown in Figure 5. [18] Inputs to DDP to generate the visualization charts are risks, mitigations to the risks, and associated weightings for risk and risk mitigations. Each of the areas of risk can be reviewed through the tool's drill-down capabilities.

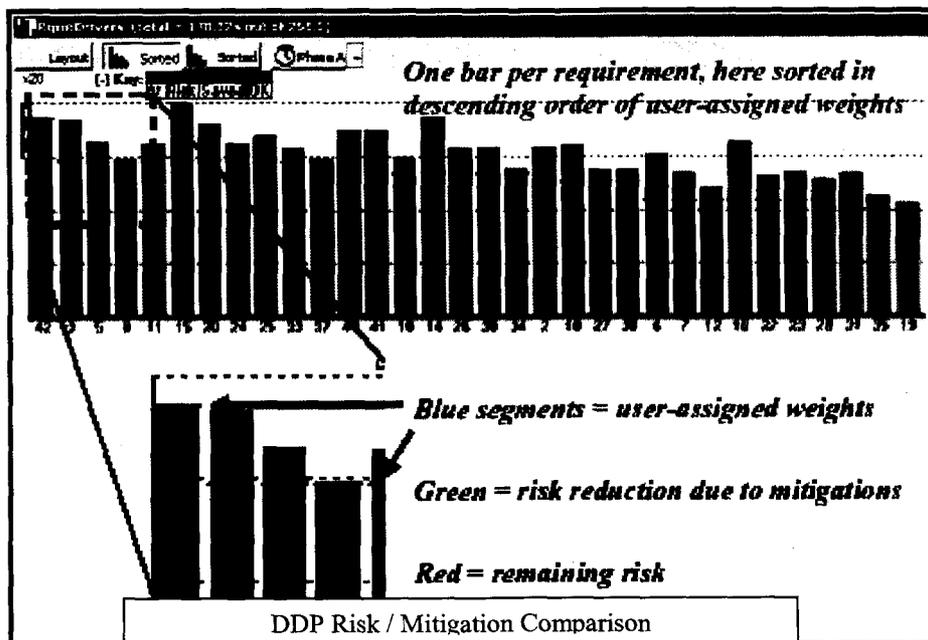


Fig. 5: Graphical presentation of risks sorted into descending order of risk levels

Identification of risks and their mitigations is a critical activity when addressing the matrix of institutional and project risk mitigations and the impact of the risk requirements and various mitigations. Examples 1 and 2 describe the impact and why it is important on an enterprise scale to use a tool such as DDP to help identify the areas of concern.

Example 1: Consider the following situation: As a risk mitigation, an institutional requirement states that all systems importing/exporting data shall have Anti-Virus (AV) software protection where available for the installed operating system. The AV software shall perform real-time virus scanning of content. These two requirements protect the institution against malicious code attacks. A project has a server that performs a large number of file transfers and is required to provide upload/download of files in near real-time. The negative impact of the real-time AV scanning on CPU cycles and the ability of the system to process file requests within the specified timeframe must be identified and addressed. Recognition of this potential conflict and alternative means to address it is facilitated early in the life cycle through the use of the risk management process described above. In this example, an alternative, previously identified mitigation would be the installation and maintenance of an AV gateway server that processes all file downloads to the server. Each of the mitigation alternatives has a cost factor—one on availability of CPU cycles, the other in equipment and support. Both factors need to be weighted for the trade-off analysis. [19]

Example 2: An institutional firewall has a positive impact in mitigating some risks in attack scenarios by preventing external port exploits. However, the firewall packet inspection may have an impact on a project requiring high throughput availability for its data. This problem can be compounded in a distributed enterprise that must pass data across the Internet. The problem is further compounded if data encryption is required by the organization. These factors must be carefully weighed and balanced. The risks must first be identified and mitigations to them provided to the project so that they can be included in the project requirements specifications. A project systems engineer may be unaware of the institutional requirement. Use of a risk management tool can help identify these types of issues for the environment.

Advantage of DDP: DDP provides the capability to semi-automate the risk management process. It also provides the ability to track risk and to update the assessment as the requirements and environment change. Auditing security risk and mitigation processes is significantly aided by using a risk management tool like DDP. Management decisions and traceability for those decisions as well as their impact can be made more easily with the risk analysis available to them. A rollup for risks over subsystems, systems, projects, and at the institutional level can be plotted using DDP to provide a management view of the overall state of risk. [20] For the life cycle, a risk baseline can be maintained. As the project's requirements and needs change over time, and as better mitigation tools are identified, the new technology and potential new threats can be compared against the baseline. This is especially useful during the SDLC maintenance phase.

4. IT Security Risk Management in the Project Life Cycle

In the project life cycle, the focus is on integrating security in the production of goods, such as software systems, and services such as Internet services (web or email hosting). Project resources may be spread across several systems which may not be co-located—potentially spread across large distances over the Internet. The System Administrator may or may not know the content or purpose of data supported by the systems. This is especially true for systems hosting distributed file sharing and enterprise project tools, including document libraries, requirements management tools, and groupware such as application development tools. Communication between project stakeholders and management is essential in the area of project life cycle security risk management. A process for addressing IT security risk in the SDLC is a requirement imposed on US federal agencies. [21] The Federal Information Processing Standards Publication, "Guidelines for Security of Computer Applications," gives guidance in identifying risks in the SDLC. However, it does not address give guidance on identifying and mitigating risks. [22]

4.1. Security Risk Management and System Engineering

Managing security risk in the project life cycle is a critical function for the enterprise as a number of the applications that are developed by the organization will be used within the institutional environment. Further, organizations that develop applications for public use or profit infuse the technology into their own

environment.[3] This practice *does* provide additional testing and validation of the software as the organization will likely be the first to experience any problems associated with an application or system. However, there may be an adverse impact in that an immature product or one that has not been carefully controlled and managed could pose a high risk to the environment. Engineering risk assessment and management in the project life cycle becomes even more critical in these cases. A recent report identifies the fact that “most operating systems and commercial software applications are known to have more than 2 defects per KSLOC, or 2,000+ defects per million SLOC.” If security defects or concerns comprise only 5% of these defects, the report explains, there are still 100 security related defects per million SLOC. [23] Since major applications and operating systems are even more extensive, this number can be a significantly larger. For middleware (also referred to ‘glueware’—software interfaces between applications), the security defect rate may be even higher.

4.2. Managing IT Security in the Project and SDLC Life Cycle

The value of SSE and risk management for the SDLC is that it brings together domain experts to address risk early on. Due to the fact that most IT environments are highly volatile, risk management of the SDLC must be a persistent process. The IT environment changes over time which affects risks and mitigations either positively with new tools, instruments and processes, or negatively such as when there is a major organizational change. The phases for coding, testing, validation, operations and maintenance must continue to have risk assessment performed. Formal tools to mitigate security risks have been developed at JPL to address the SDLC. These tools provide a unified approach to addressing SDLC vulnerabilities and exposures and are being integrated into DDP to provide an SDLC security risk management process. The approach includes a Security Checklist (SC), a vulnerability matrix, model-based verification, property-based testing, a list of Security Assessment Tools (SAT), and training. A two-phased SC that addresses the project life cycle and the external release of software has already been developed for NASA. [24] The SC identifies critical areas of risk in the SDLC that need to be addressed. The SC includes verification and validation of requirements as they flow from specification through the design, development, operations and maintenance phases. A vulnerability matrix that classifies vulnerabilities and exposures, and a list of security assessment tools is currently being maintained by the University of California at Davis. [25] Figure 6 depicts the use of these tools in the SDLC and a unified process for risk management/mitigation.

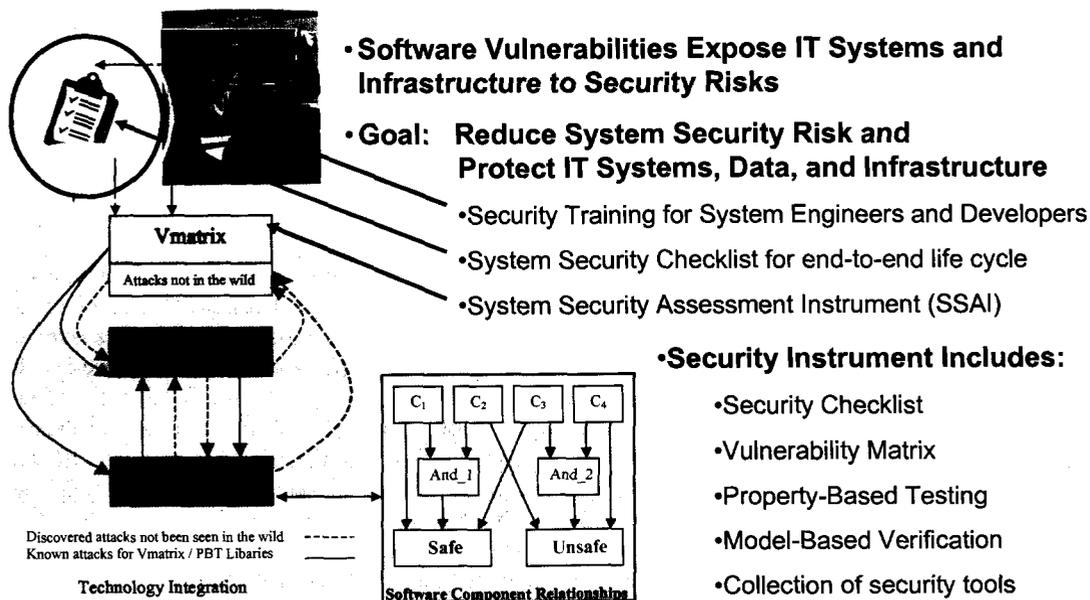


Fig. 6: Unified project life cycle risk mitigation approach

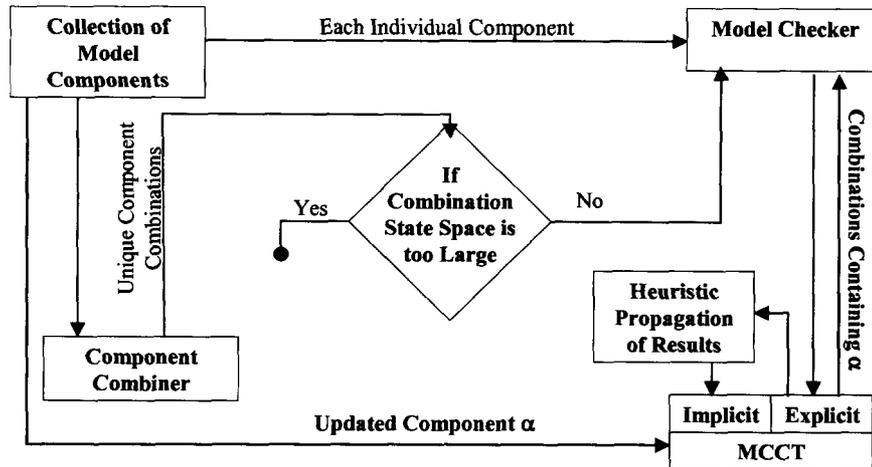


Fig.7: Model component and combination tree

4.3. Model Checking and the Flexible Modeling Framework

Recently delivered to NASA is a software Model Checking (MC) technique for use in the requirements specification phase of the SDLC, the Flexible Modeling Framework (FMF). MC offers the benefit by identifying problems early in the SDLC where it is less costly and easier to correct them. MC provides a formal analytical approach for integrating security into existing and emerging practices for developing high quality software and computer systems. MC can identify vulnerabilities and undesired exposures in software. These often arise from a number of development factors that can often be traced to poor software development practices, new modes of attacks in the network security arena, unsafe configurations, and unsafe interaction between systems and/or their components. The most extreme scenario is when a system is connected to the Internet. MC offers a means for examining component interaction in relation to critical system properties such as safety and security. The use of MC as means of verification to mitigate vulnerabilities during the life cycle suffers from some practical limitations. Among these limitations are:

- The frequency of an exploit being perpetrated is based on three factors: how easy it is to originate
Limits on the size and complexity of systems that may benefit from MC given reasonable computer memory resources
- Difficulty in rapid development, modification and verification of models in a timely manner during the early life cycle when systems tend to change and evolve quickly.

Traditionally, software model checkers automatically explore all paths from a start state in a computational tree that is specified in an MC model. The computational tree may contain repeated copies of sub-trees. State of the art Model Checkers such as SPIN exploit this characteristic to improve automated verification efficiency. The objective is to verify system properties with respect to models over as many scenarios as feasible. Since the models are a selective representation of functional capabilities under analysis, the number of feasible scenarios is much larger than the set that can be checked during testing. FMF employs MC as its core technology and provides a means to bring software security issues under formal control early in the life cycle. [25, 26] The approach is shown in Figure 7.

The MC FMF seeks to address the problem of formal verification of larger systems by a divide and conquer approach. [27] First, by verifying a property over portions of the system, then incrementally inferring the results over larger subsets of the entire system. As such, the FMF is: 1) a system for building models in a component based manner to cope with system evolution over time and, 2) an approach of compositional verification to delay the effects of state space explosion. This methodology allows property verification results of large and complex models to be examined and extrapolated appropriately.

Modeling in a component-based manner involves building a series of small models, which later will be strategically combined for system verification purposes. This strategic combination correlates the modeling function with modern software engineering and architecture practices whereby a system is divided into major parts, and subsequently into smaller detailed parts, and then integrated to build up a software system. An initial series of simple components can be built when few operational specifics are

known about the system. However, these components can be combined and verified for consistency with properties of interest such as software security properties.

4.4. Property Based Testing

Property Based Testing (PBT) is different from formal verification. It recognizes that implementation difficulties, and environment considerations, may affect conformance to the properties (and hence the security of execution). A key observation is that testing does not validate that a program will always meet the properties, unless all possible paths of execution are traversed. But it does provide additional assurance that the implementation is correct, and does satisfy the properties, when execution follows the tested control and data flow paths.

Many control and data flow paths are irrelevant to the program's satisfying the desired properties. A technique called slicing [28] creates a second program that satisfies the properties if, and only if, the original program satisfies those properties. The second program contains only those paths of control and data flow that affect the properties. This focuses the testing on paths of execution relevant to the security properties, rather than on all possible paths of execution (See Figure 8).

The figure below captures the PBT verification process: given a knowledge of security and an accurate specification of the security model (which says what is and is not allowable), the software is analyzed to determine the level of assurance, or belief that the program does what it is intended to do. The properties being tested are directly taken from the security properties of the model. The expectation is that the code honors these. The program is then *sliced* to derive the smallest program equivalent to the original with respect to the stated properties. The program is then instrumented and tested. The testing either validates the properties or shows the do not hold. The tester helps determine the level of assurance of the program

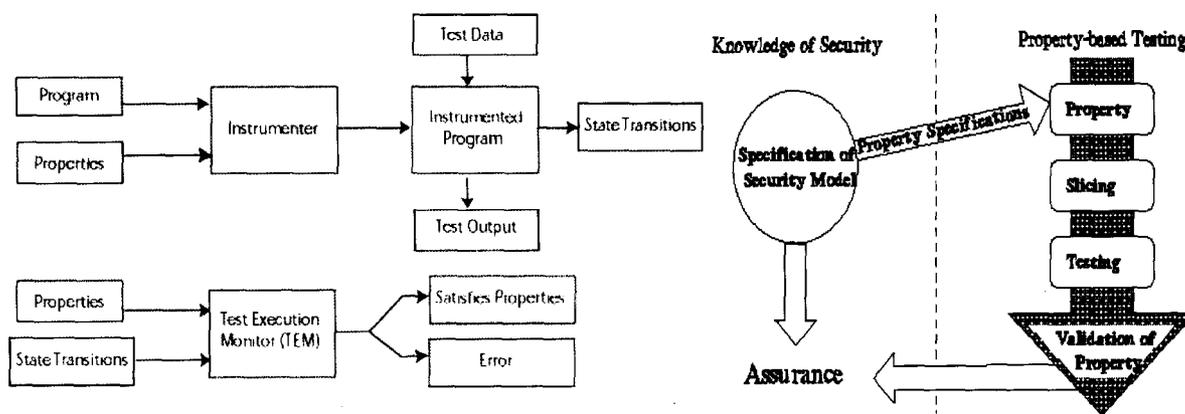


Fig. 8: PBT Process and Model

Implementation and operations also have special needs attached as well, such as removal of installation files which can be used to overwrite current configuration settings, or leaving configuration settings in an unsecured state after installation (usually settings are left at the default which generally has few security controls). Assignment of personnel responsibilities and setting up accounts and access control lists to the system and data is another issue in this phase. In particular, the maintenance phase is where a number of problems can arise where hardware and/or software is replaced or patched. When modules are replaced, the modules and interacting modules, at a minimum must be re-verified, and the system itself must be re-validated to process data. Often modifying the original system can inadvertently create vulnerabilities or unwanted exposures. For example, some modules that previously had been tested and verified as 'safe'. When they receive input that has changed due to changes in another module, a potential for an unintended weakness may now exist. Additionally, documentation must be updated to reflect the change, particularly when it affects operations and operational processes. When decommissioning a system to which another system has a dependency it may leave the related system in a vulnerable state. For example, an open port may exist which is waiting for data from a non-existent system. This is a high-risk problem as it provides

an avenue for compromise. Performing a risk assessment whenever there is a significant change to the system environment especially when a network aware application or a system is shut-down is essential. Use of the described modeling and property based testing are useful preventatives in mitigating these risks.

5. IT Security Risk Management in the Institution

Managing IT security institutionally must be an enterprise-wide effort. Not only does it require management support, but it also needs to be coordinated with the project life cycle. There is a mutual impact for managing and mitigating risks. Paramount is identification of IT resources, data and processes with the goal of protecting them. Results of an institutional risk assessment at JPL show that the following activities provide a high degree of mitigation at a favorable cost to the entire organization: [29]

- Use of an IT Security Database (ITSDB)
- Ability to test, disseminate and apply patches quickly (either automated or semi-automated)
- Use of an intrusion detection system (IDS) that monitors traffic on the network
- Scanning systems to verify that IT computer systems are not vulnerable to attacks
- An automated security problem tracking system to ensure that systems are not vulnerable
- Firewalls

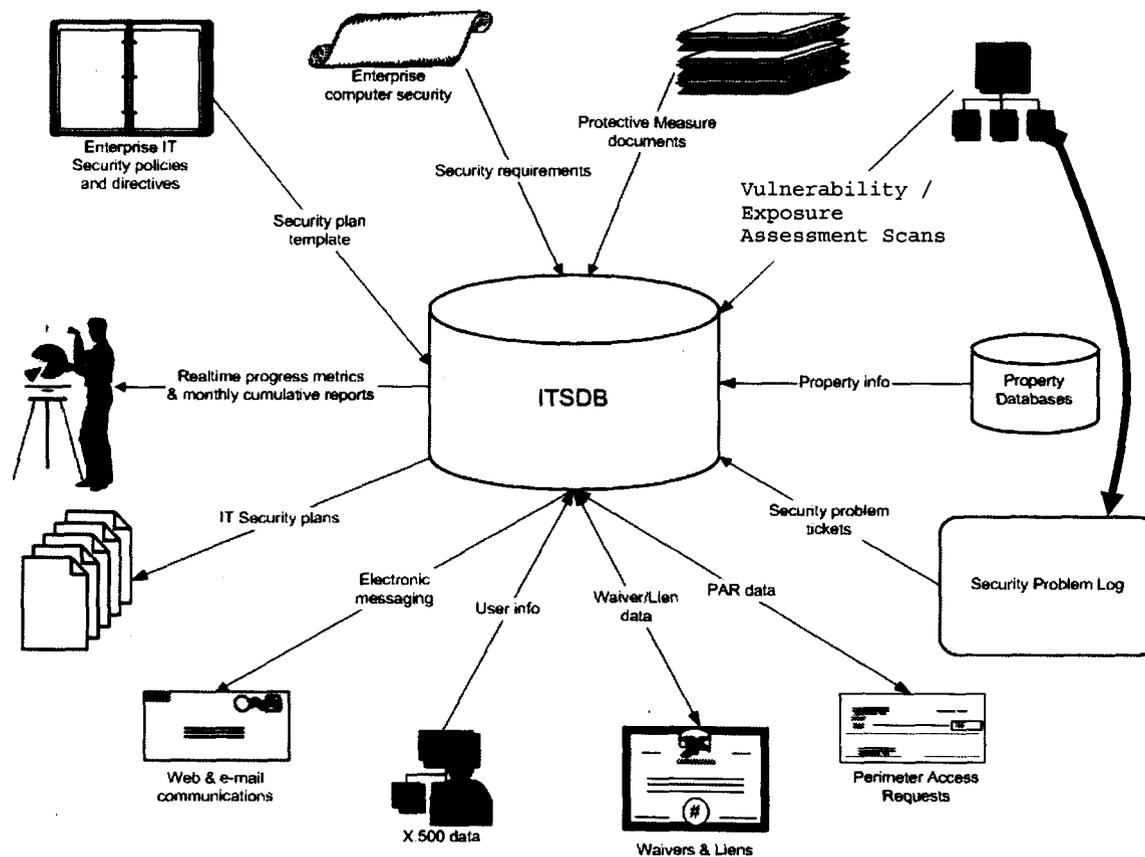


Fig. 9: IT Security Plan Database (ITSDB)

To manage a process of this magnitude, use of an IT Security Database (ITSDB) containing the elements to be managed facilitates the ability to automate risk assessment and mitigation activities. Identification of IT resources and collecting the information in an ITSDB facilitates the ability to manage these resources. The ITSDB should include processes that provide input into it to capture requirements from the various sources such as standards, guidelines, policies, etc. [30, 31] It can also serve as the focal point for centrally controlling management of IT risk, risk mitigations as well as IT resources. Without

automated processes and control points, it is difficult at best to manage IT security at the enterprise level. An institutional approach to managing IT security risk at the enterprise level is shown in Figure 9 where the ITSDB and risk management process identifies and controls security risks. [29] Results of the analysis feed into processes that provide input and receive output from the ITSDB.

An ITSDB should include the following major elements: A security plan template for aiding in writing security plans for IT systems and resources, IT security policies and requirements for identifying lines of responsibilities, system and data protection, protective measures for operating systems, user, system administrator, cognizant management assigned responsibility for the IT resources, configuration control information for critical assets, identified security problems for any IT resources. An ITSDB process should also output metrics, produce plans from the information entered, have email communication processes that alert managers and cognizant personnel to address security problems identified with the IT resources. JPL has successfully instituted an ITSDB and its associated process as described here along with the other technologies (patching services, IDS and Scanning, Security Problem Log, and Firewalls among other technologies—see Figure 9 - ITSDB).

In the ITSDB management process, enterprise IT security policies and directives require the implementation and control of security plans for the IT resources. These are formalized into requirements and protective measures, in particular protective measures for computer systems. This practice allows for the auditing of systems based on requirements and implementations that meet those requirements, including the continued updating of systems to meet the requirements for a secure computing system as new vulnerabilities are discovered.

5.1. Security Problem Log (SPL) Database

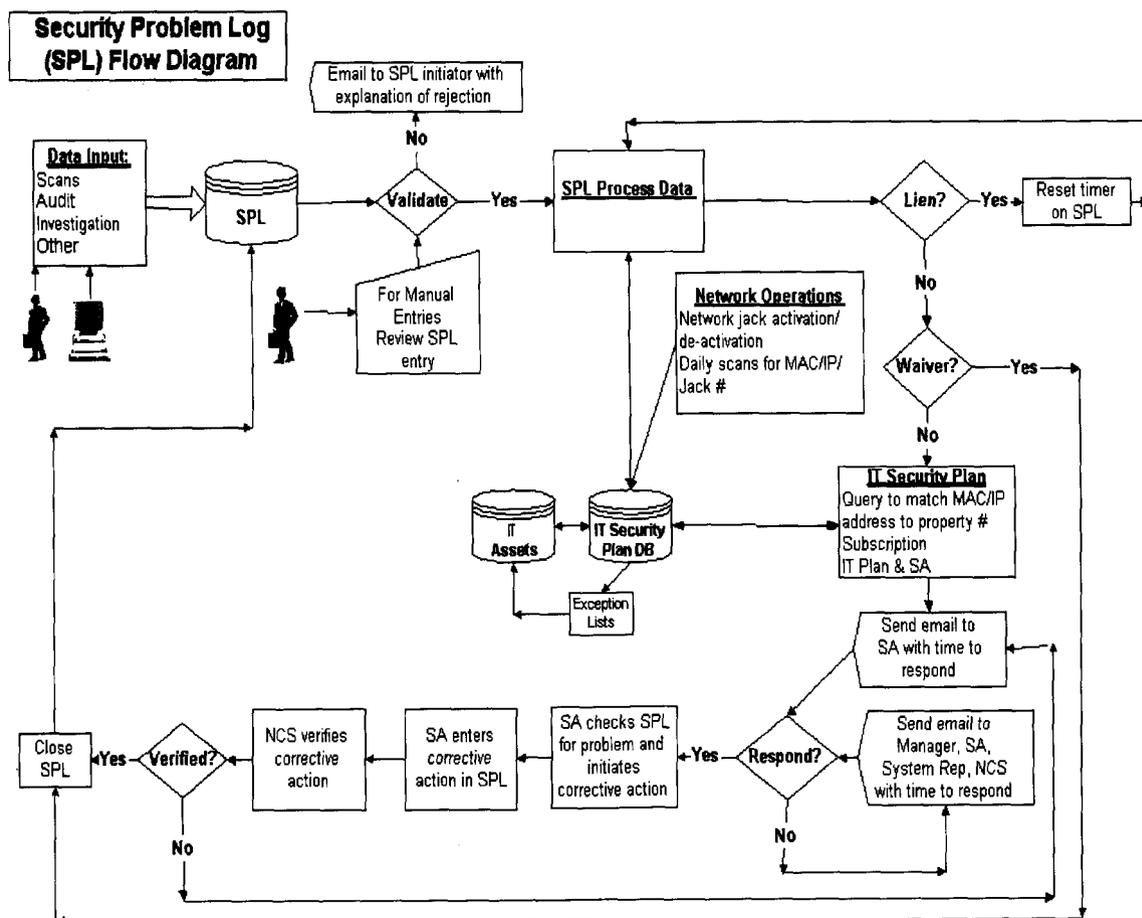


Fig. 10: Security Problem Log (SPL)

Another security risk control/mitigation system developed and used at JPL is a security problem tracking system to ensure that patches are applied to systems within a specified timeframe. The JPL ITSDB is integrated with a Security Problem Log (SPL) database (Figure 10). The SPL maintains records of security problems identified for IT systems. If a system has an identified vulnerability, an automated process creates an SPL ticket and emails are sent to key people to take corrective action. The SPL ticket identifies a time-frame to fix the vulnerability, usually through the application of a patch. If a security problem ticket is not closed within the specified time-frame, the problem is escalated. If the problem persists without corrective action being taken, the system may have network access blocked.

The JPL SPL has the following elements and processes:

- Records security problems found during internal on-site audits and electronic vulnerability scans
 - Issues tickets to be closed by expiration date
 - Notifies responsible personnel by email, with escalation to line managers for expired tickets
- Supports viewing and updating SPL tickets
 - Provides detailed vulnerability description and specific instructions for the preferred corrective action
 - One-click corrective action response, if preferred fix was applied
 - Accommodates rejection requests and requests to delay expiration date (liens)
 - Prevents creation of new tickets when there are previous false positives, waivers, liens, or duplicate open tickets
 - Closes open tickets when corrective action is verified, if a waiver is approved, or a false positive is confirmed

6. Conclusion

These risk management activities at JPL that address both institutional and project life cycles have shown that formalizing the process is highly effective and beneficial to both. Effective risk management must be a proactive and persistent activity that involves the organization at both the institutional and the project levels. It requires the cooperation of everyone in the organization. An SSE can manage security risks by working with domain experts and management to identify the risks and their mitigations. Use of a risk management tool can help provide objective control of the risk elements and their interactions both institutionally and in the SDLC.

While the institutional risk mitigation processes may benefit the life cycle, they must be carefully weighed and balanced against other risks and the potential impact of the mitigations, especially in the interface with the project life cycle. Institutional risk abatement activities for the enterprise provide mitigations for the project life cycle and should be accounted for as part of the risk assessment and mitigation analysis process. Further, Integrating risk mitigations provided by the institution into the project life cycle helps to identify risks that may already be costed independently. The projects may rely on institutional mitigation for risks identified in its own processes which could reduce its overall security risk mitigation costs while providing higher security as well. Consequently, some of the mitigations, even though more costly when provisioned independently, may actually be cheaper as the costs are shared across the organization and are already factored into the project costs for institutional support. For this additional reason, it is more cost effective to implement an institutional risk assessment and mitigation program as described above. Spreading the cost of providing risk mitigation across projects actually reduces the cost for each project of providing its own support and tools independently.

Applying a risk management process to IT security is a critical activity to prevent loss or compromise of CIA. An overall architecture to manage IT security risk enables organizations to understand these risks better, including the likelihood of success, the potential for damage if successful, the effectiveness and cost of mitigations. It gives managers the capability to make informed decisions on mitigating risk and accepting residual risk, along with the associated costs. Such a methodology applied as a systems engineering practice both institutionally and in the SDLC at the project level enables the organization to respond quickly and more effectively to new threats as the environment and technology change over time. For both the institution and projects performing risk assessment as part of an IT security plan process helps the organization to understand the security needs of the organization and provide the capability for full-cost accounting for both the institution and the project. The risk management activities identified above have benefited JPL in its efforts to take proactive and cost effect steps in protecting the organization.

Acknowledgement

The research described in this paper was carried out at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government or the Jet Propulsion Laboratory, California Institute of Technology. Contributions from Martin Feather and Sharon Hope, are recognized and appreciated. They have been valuable in providing feedback for the concepts presented in this paper.

References

1. Anderson, R. J., Security Engineering: A Guide to Building Dependable Distributed Systems, John Wiley & Sons; (2001)
2. GAO-03-98, Government Accounting Office (GAO) Audit: "Major Management Challenges and Program Risks: Department of Defense," GAO-03-98, January 2003 (available on the Internet at: <http://www.gao.gov/pas/2003/>)
3. Howard, M. and LeBlanc, D., Writing Secure Code, Second Edition, Microsoft Press, December, 2002
4. NASA CRM Resource Center website <http://www.crm.nasa.gov/knowledge/default.html>, accessed 09-15-2003
5. Stamatelatos, M.G., "Risk Assessment and Management, Tools and Applications," PowerPoint Presentation, available on NASA CRM Resource Center website: http://www.crm.nasa.gov/papers/presentation_1.pdf (accessed 09-20-03)
6. Witty, R., "Successful Elements of an Information Security Risk Management Program," Gartner Symposium ITxpo, U.S. Symposium/ITxpo, Orlando, Florida, 6-11 October 2002.
7. ArcSight, "TruThreat Visualization Software", 2003, available at <http://www.arcsight.com/>.
8. RiskWatch, "Security Risk Management (SRM) software solutions for government and industry", information downloaded from the Internet on 10-10-03, <http://www.riskwatch.com/>.
9. McGraw, G., "Software Risk Management for Security", Citigal White Paper, 1999, available at <http://www.cigital.com/whitepapers/>.
10. ISO, International Organization for Standardization, ISO 9000:2000 family, Quality Management Systems, http://www.iso.ch/iso/en/iso9000-14000/iso9000/selection_use/iso9000family.html, accessed 09-19-2003
11. Carnegie Mellon University (CMU) Software Engineering Institute (SEI) Capability Maturity Model® Integration (CMMISM), available on the Internet at <http://www.sei.cmu.edu/cmmi/general/> (accessed 09-20-03).
12. SEI, Carnegie Mellon University Software Engineering Institute, "OCTAVE Method," 11-11, 2003, available at <http://www.cert.org/octave/methods.html>.
13. NIH CIT (National Institute of Health, Center for Information Technology), "NIH Application/System Security Plan Template for Major Applications and General Support Systems," 1994.
14. Bishop, M., "Computer Security: Art and Science", Addison-Wesley Pub Co.; (2002).
15. Feather, M.S., Cornford, S. L., and Moran, K., "Risk-Based Analysis And Decision Making In Multi-Disciplinary Environments," Proceedings of IMECE'03 2003 ASME International Mechanical Engineering Congress & Exposition Washington, D.C., November 16-21, 2003.
16. Cornford, S. L., Feather, M. S., and Hicks, K. A., "DDP - A tool for life-cycle risk management," IEEE Aerospace Conference, March 2001, (available on the web at: <http://ddptool.jpl.nasa.gov>)
17. Cornford, S. L., Feather, M.S., Dunphy, J., Salcedo, J., and Menzies, T., "Optimizing Spacecraft Design - Optimization Engine Development: Progress and Plans," IEEE Aerospace Conference, March 2003, (available on the web at: <http://ddptool.jpl.nasa.gov>)
18. Feather, M.S., Cornford, S. L., and Dunphy, J., "A Risk-Centric Model for Value Maximization," Proceedings, 4th International Workshop on Economics-Driven Software Engineering Research, Orlando, Florida, May 21 2002, pp. 10-14

19. Feather, M.S., Hicks, K.A., Johnson, K.R., and Cornford, S. L., "Software Support for Improving Technology Infusion," Proceedings of the 1st International Conference on Space Mission Challenges for Information Technology (SMC-IT), Pasadena, California, July 2003, pp. 359-367; JPL Publication 03-13A, Jet Propulsion Laboratory, California Institute of Technology
20. Cornford, S., "Defect Detection and Prevention (DDP): A Tool for Life Cycle Risk Management: Explanations, Demonstrations and Applications," DDP Tool Training Seminar presented at JPL at the Jet Propulsion Lab, March 23, 2001
21. Swanson, M., "Guide for Developing Security Plans for Information Technology Systems," NIST Special Publication 800-18, 1998
22. FIPS PUB 73, Federal Information processing Standards Publication, "Guidelines for Security of Computer Applications," 1980
23. Heinz, L., "Preventing Security-Related Defects," news@sei interactive, 2Q, 2002, downloaded from the Internet at: <http://interactive.sei.cmu.edu>, August 19, 2003.
24. Gilliam, D., Wolfe, T., Sherif, J., and Bishop, M., "Software Security Checklist for the Software Life Cycle," Proc. of the Twelfth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, Linz, Austria, pp 243-248.
25. Gilliam, D., Kelly, J. Powell, J., Bishop, M., "Development of a Software Security Assessment Instrument to Reduce Software Security Risk" Proc. of the Tenth IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises, Boston, MA, pp 144-149.
26. D. Gilliam, J. Powell, J. Kelly, M. Bishop, "Reducing Software Security Risk Through an Integrated Approach," 26th International IEEE/NASA Software Engineering Workshop, 17-29 November, Greenbelt, MD.
27. Component Based Model Checking, J. Powell, D. Gilliam, Proceedings of the 6th World Conference on Integrated Design and Process Technology, June 23-28, Pasadena CA, p 66 & CD
28. Weiser, M., "Program Slicing," IEEE Transactions on Software Engineering SE-10(4) pp. 352-357 (July 1984).
29. Miller, R. L., "JPL's Infrastructure for Managing IT Security: The Processes and Custom Toolset," presentation to the NASA IT Security Managers' Workshop, April, 2003.
30. Stoneburner G., Goguen, A., and Feringa, A., "Risk Management for Information Technology Systems," The National Institute of Standards and Technology Special Publication 800-30, 2001
31. Stoneburner, G., Hayden, C., and Feringa, A., "Engineering Principles for Information Technology Security (A Baseline for Achieving Security)," NIST Special Publication 800-27.

Abbreviations

CIA	Confidentiality, Integrity and Availability
DDP	Defect Detection and Prevention
FMF	Flexible Modeling Framework
ITSDB	Information Technology Security Database
KSLOC	Thousand Source Lines of Code (or SLOC—Source Lines of Code)
MC	Model Checking
PBT	Property Based Testing
SAT	Security Assessment Tools
SDLC	Systems Development Life Cycle
SC	Security Checklist
SE	Security Engineering
SPL	Security Problem Log
SSE	System Security Engineer



Mitigations

L [Effects] Add [Icons]

300	<input type="checkbox"/>	1:Use of VPN
200	<input type="checkbox"/>	2:Institutional firewall
30	<input type="checkbox"/>	3:SysAdmins keep patches up to date
100	<input type="checkbox"/>	4:Train users on selection of passwords
200	<input type="checkbox"/>	5:Use an IDS system and respond to its alerts
500	<input type="checkbox"/>	6:Keep critical data encrypted
300	<input type="checkbox"/>	7:Host-based firewalls
400	<input type="checkbox"/>	8:Security scans

Mitgn x Risk [-1...+1]: 0 or empty = no effect; 1 = 100% effect

Layout [Icons] Num Edit [Dropdown] [Warning Icon] (Off)

Mitgn x Risk Col = Buffer overflows
Row = Security scans

	Risks	simple	sophis	Open	malicic	malicic	Buffer	Non-cc
Mitgn counts	8	7	6	6	4	6	8	
Use of	3	0.8	0.8				0.8	
Instituti	7	0.5	0.5	0.6	0.5	0.1	0.3	
SysAdm	6	0.1	0.1	0.9	0.8		0.95	
Train	4	0.9	0.7		0.2		0.4	
Use	7	0.2	0.2	0.4	0.9	0.8	0.75	
Keep	7	0.5	0.5	0.6	0.1	0.95	0.95	
Host-b	7	0.95	0.95	0.7	0.2	0.1	0.5	
Securit	4	0.95		0.95			0.95	

Risks

L [Effects] Add [Icons]

<input checked="" type="checkbox"/>	1:simple password hacking
<input checked="" type="checkbox"/>	2:sophisticated password hacking
<input checked="" type="checkbox"/>	3:Open ports availability
<input checked="" type="checkbox"/>	4:malicious website code allows download of SAM DB
<input checked="" type="checkbox"/>	5:malicious code that passes information off to another location
<input checked="" type="checkbox"/>	6:Buffer overflows
<input checked="" type="checkbox"/>	7:Non-compliant users



Risks

Effects Add [checkbox] [checkbox] [checkbox]

of 7: Host-based firewalls

0.95	<input checked="" type="checkbox"/>	1:simple password hacking
0.95	<input checked="" type="checkbox"/>	2:sophisticated password hacking
0.7	<input checked="" type="checkbox"/>	3:Open ports availability
0.2	<input checked="" type="checkbox"/>	4:malicious website code allows downloa
0.1	<input checked="" type="checkbox"/>	5:malicious code that passes information
0.5	<input checked="" type="checkbox"/>	6:Buffer overflows
0.5	<input checked="" type="checkbox"/>	7:Non-compliant users

Objvs: 0 out of 56

Layout (All)

TOTAL \$0 (of which \$0 are repair costs)

Checked	Cost
	0

Mitigation viewer

Layout (All)

1:simple password hacking

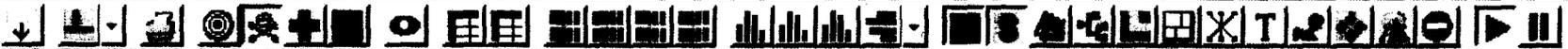
#	Effect	Cost	Title
<input checked="" type="checkbox"/> 7	0.95	300	Host-based firewalls
<input type="checkbox"/> 8	0.95	400	Security scans
<input type="checkbox"/> 4	0.9	100	Train users on selection of passw...
<input type="checkbox"/> 1	0.8	300	Use of VPN
<input type="checkbox"/> 6	0.5	500	Keep critical data encrypted
<input type="checkbox"/> 2	0.5	200	Institutional firewall
<input type="checkbox"/> 5	0.2	200	Use an IDS system and respond t...
<input type="checkbox"/> 3	0.1	30	SysAdmins keep patches up to date

RiskBalance (log scale)

Tree Order Hold Sort 10 1 RISK: APriori Set Higher Lower Categories General

Risks 1:simple password hacking

Item	Risk Level
1	0.95
2	0.95
3	0.7
4	0.2
5	0.1
6	0.5
7	0.5



Risks

Effects Add [Icons]

of 7: Host-based firewalls

0.95	<input checked="" type="checkbox"/>	1:simple password hacking
0.95	<input checked="" type="checkbox"/>	2:sophisticated password hacking
0.7	<input checked="" type="checkbox"/>	3:Open ports availability
0.2	<input checked="" type="checkbox"/>	4:malicious website code allows downloa
0.1	<input checked="" type="checkbox"/>	5:malicious code that passes information
0.5	<input checked="" type="checkbox"/>	6:Buffer overflows
0.5	<input checked="" type="checkbox"/>	7:Non-compliant users

Objys: 0 out of 56

Layout (All)

TOTAL \$300 (of which \$0 are repair costs)

	Cost
Checked	300

Mitigation viewer

Layout (All)

1:simple password hacking

#	Effect	Cost	Title
<input checked="" type="checkbox"/> 7	0.95	300	Host-based firewalls
<input type="checkbox"/> 8	0.95	400	Security scans
<input type="checkbox"/> 4	0.9	100	Train users on selection of passw...
<input type="checkbox"/> 1	0.8	300	Use of VPN
<input type="checkbox"/> 6	0.5	500	Keep critical data encrypted
<input type="checkbox"/> 2	0.5	200	Institutional firewall
<input type="checkbox"/> 5	0.2	200	Use an IDS system and respond t...
<input type="checkbox"/> 3	0.1	30	SysAdmins keep patches up to date

RiskBalance (log scale)

Tree Order Hold Sort 10 1 RISK: APriori Set Higher Lower Categories General

Risks 1:simple password hacking

Item	Risk Level
1	0.95
2	0.95
3	0.7
4	0.2
5	0.5
6	0.5
7	0.1



Risks

Effects Add [Icons]

of 3: SysAdmins keep patches up to date

0.1	<input checked="" type="checkbox"/>	1:simple password hacking
0.1	<input checked="" type="checkbox"/>	2:sophisticated password hacking
0.9	<input checked="" type="checkbox"/>	3:Open ports availability
0.8	<input checked="" type="checkbox"/>	4:malicious website code allows downloa
	<input checked="" type="checkbox"/>	5:malicious code that passes information
0.95	<input checked="" type="checkbox"/>	6:Buffer overflows
0.5	<input checked="" type="checkbox"/>	7:Non-compliant users

Objvs: 13.1 out of 56

Layout (All)

TOTAL \$330 (of which \$0 are repair costs)

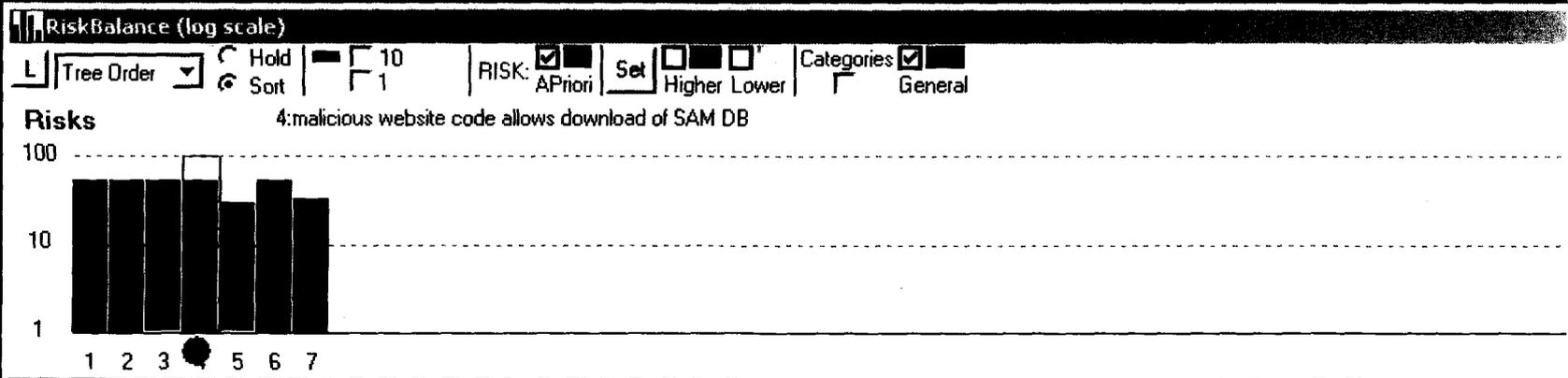
Cost
Checked 330

Mitigation viewer

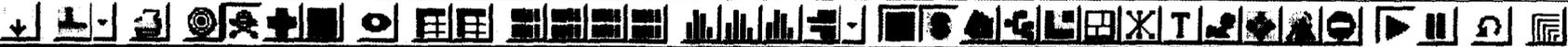
Layout (All)

4:malicious website code allows download of SAM DB

#	Effect	Cost	Title
<input type="checkbox"/> 5	0.9	200	Use an IDS system and respond t...
<input checked="" type="checkbox"/> 3	0.8	30	SysAdmins keep patches up to date
<input type="checkbox"/> 2	0.5	200	Institutional firewall
<input type="checkbox"/> 4	0.2	100	Train users on selection of passw...
<input checked="" type="checkbox"/> 7	0.2	300	Host-based firewalls
<input type="checkbox"/> 6	0.1	500	Keep critical data encrypted



File Options Window Help



Risks

Effects Add [checkbox] [checkbox] [checkbox]

of 5: Use an IDS system and respond to its alerts

0.2	<input checked="" type="checkbox"/>	1:simple password hacking
0.2	<input checked="" type="checkbox"/>	2:sophisticated password hacking
0.4	<input checked="" type="checkbox"/>	3:Open ports availability
0.9	<input checked="" type="checkbox"/>	4:malicious website code allows downloa
0.8	<input checked="" type="checkbox"/>	5:malicious code that passes information
0.75	<input checked="" type="checkbox"/>	6:Buffer overflows
0.4	<input checked="" type="checkbox"/>	7:Non-compliant users

Objvs: 39.3 out of 56

Layout (All)

TOTAL \$530 (of which \$0 are repair costs)

Cost
Checked 530

Mitigation viewer

Layout (All)

5:malicious code that passes information off to another location

#	Effect	Cost	When	Title
<input type="checkbox"/> 6	0.95	500		Keep critical data encrypted
<input checked="" type="checkbox"/> 5	0.8	200		Use an IDS system and respond t...
<input checked="" type="checkbox"/> 7	0.1	300		Host-based firewalls
<input type="checkbox"/> 2	0.1	200		Institutional firewall

RiskBalance (log scale)

Tree Order Hold Sort 10 1 RISK: APriori Set Higher Lower Categories General

Risks 5:malicious code that passes information off to another location

Category	Risk Level (log scale)
1	~30
2	~30
3	~30
4	~30
5	~30
6	~30
7	~30