Secure Coding.
Practical steps to defend your web apps.

SANS Software Security FAQ

Version 1.0 - January 2008

Project Director	Tanya Baccam
Project Lead	Leo McCavana
Authors	Tanya Baccam Ralf Durkee Barbara L. Filkins Kevin Fuller Leo McCavana Mark Williams Lenny Zeltser

Version	Date	Publication Notes
1.0	01/27/08	Initial publication

Table of Contents

Foreword by Stephen Northcutt

Software Security comes down to two basic things, adopting a development methodology to minimize the number of vulnerabilities that get baked into the software and two, rigorous testing to find the problems that do exist. One day when we have the first approach down to a true science we will not need the second, or at least not to the extent we do today, however, for the next ten years at least, we will need robust testing. Where do we need to engineer good process and test for problems the most? Every time there is input to our program, especially from any external source, even another program, we are at risk. We need to design carefully to manage and validate input and we need to test these points in our software rigorously. But what is the second most important thing to consider?

I asked Ivan Arce of CORE Security what he felt was right up there with input validation. He said, "*Trust! So many instances of trust problems are with the interactions between applications or different components of the same application, or the user and the application. Just think about the recent problem with Acrobat Reader; you can embed a URL into a .pdf and then it might process it. If the document is on the web, Internet Explorer might then try to figure out which operating system component is needed to process it and pass the URI to that component, but if the input is not sanitized problems can arise. Internet Explorer trusts the operating system to handle the URI with the same care that Internet Explorer uses, but that may not happen. You can think of this as a chain of trust.*

Two tips, be a bit more paranoid, adopt defensive coding; do not trust every other component of the application. Instead, assume they will fail or that they are the opponent, they will try to break your application. Then you put checks and balances, or security in depth into the part of the code that you write.

Second, be explicit about how you are supposed to operate with your component and what security assumptions you are making. Be clear to help other people not make mistakes with their code. Create explicit, understandable interfaces that make the security assumptions clear."

I don't know about you, but those words give me chicken skin (Hawaiian for goose bumps) every time I read them. So where do you start on your software security journey? Tanya Baccam led a team as part of the Cyber Defense Initiative to develop this FAQ. They did a great job and we owe them a debt of gratitude. The Software Security FAQ is free, it is available, and if you can improve it, please send your improvements to stephen@sans.edu and we will review it and if we can accept it, we will add it to the work. We have also just finished working on a Software Security Awareness course that runs two to three hours. If you want that information taught in your organization, you know where to find us.

Stay Safe!

Stephen Northcutt, President

SANS Technology Institute, a postgraduate security college

Introduction

This document presents best practice answers to frequently asked questions in relation to software security. The document is targeted to those implementing software security controls within their environment. The structure of the document focuses broadly on security issues that are relevant to any stage of a typical software development life cycle.

This document also presents best practice advice relating to a variety of activities and disciplines that have an impact upon software security. To this end, coverage is given of issues such as:

- Using vended and open source software more securely.
- The need for security training and awareness – not just for developers but also managers and leaders.
- The need to have clearly devised security roles and responsibilities that are sensitive to individual project and organizational complexities.
- The importance of policy and compliance to improving the quality of software security.

This document also reflects the needs of non-technical readers (e.g. managers and leaders) and those perhaps new to the concept of software security in general. In this respect, in addition to providing some "Basic Definitions" associated with software security, the "Starting Out" section is useful to those who either need a business justification for being concerned with software security, or perhaps need advice on developing a software security program.

Section 1 – Basic Definitions

Section 1 – Basic Definitions

What is a software security program?

A software security program describes the approach an organization takes to implement a corporate culture change which is required to adopt software security requirements and improvements throughout the software development life cycle. There are three essential components needed to make such a corporate cultural change successful:

- Identification and assignment of security specific roles and responsibilities.
- Employee training for not just developers but also non-technical and managerial roles that are associated with a project.
- Development and implementation of processes to improve the overall quality of software from a number of perspectives including, but not limited to, security.

What is "Application Security" and how does it compare to "Software Security"?

These terms are often loosely used interchangeably within the field, so it is always wise to get clarification when discussing these topics. However, Software Security is often accepted to specifically address the idea of designing software with security in mind so that it can continue to function when under attack.

Application Security has been defined within the industry in many different ways. Although when compared to Software Security, it often addresses the security of an application after it has been built.

Section 2 – Starting Out

Section 2 – Starting Out

We use multiple platforms from AS400, right through to Web 2.0 applications. Where on earth do I start to understand our software security requirements and where do I concentrate my efforts?

The best starting point depends on the extent of your responsibilities, your decision-making power, and the amount of good will you have established with the development teams and business users.

If your responsibilities span several application platforms, begin by assessing the risks associated with each of them. Then, concentrate your initial efforts on the riskiest applications. The risk assessment should incorporate factors such as the current state of the application - how vulnerable it is - as well as the amount of money the organization would loose should the application's security be compromised.

If a comprehensive risk assessment is an impractical undertaking for you at this time, consider starting with an "easy win" - an application that is both important to the organization and that you are able to handle. Concentrating your efforts on a project that is most likely to succeed will allow you to demonstrate the benefits of improving software security, strengthening your reputation and building additional good will.

Build upon your success with the initial project to expand your efforts to other application platforms in your organization.

We have multiple business units in our enterprise, each with their own development methodologies. How do I go about developing a single 'fits all' answer that addresses everyone's needs?

A single 'fits all' answer may not exist in a heterogeneous business environment. Attempting to develop a unified methodology that addresses all concerns of everyone may be an exercise in futility. Instead, consider establishing a common high-level set of practices that are relevant to all business units. Then, recognizing the unique requirements of each group, develop a process that allows each business unit to fine-tune the baseline in line with its needs, restrictions, and capabilities. The baseline framework should refer to the organization's criteria for assessing and managing risks, to ensure that a business unit does not expose the organization to unnecessary risk.

Some business units appear to be better at software and application security than others. What should I do?

This is a great opportunity to build upon internal security knowledge, rather than having to start the learning process from scratch. Set up a process that allows the business units with stronger application security practices to share their experiences with the other teams via training sessions, newsletters, and

informal meetings. Build upon the practices that have worked well, learn upon the past mistakes, and consider whether the frameworks that these business units developed can be adapted by the other groups.

What are some methods to convince management that it is in everyone's best interest to develop a software security program?

Understand the management's concerns, learn to speak their language, and present your recommendations in a manner that is relevant to them. This often involves describing the business risks facing the organization without a software security program. Be specific regarding the money that can be lost and how much it will cost to address the risks.

Tie your recommendations to the organization's risk management program, if it exists. Provide concrete steps for establishing the software security program, rather than describing general concerns.

Additionally, consider legal, business, and customer requirements that may drive the organization to strengthening its application security program. Are your institutional clients or partners imposing security requirements? Are regulators enforcing security mandates relevant to your industry? Are state laws making it costly to deal with data compromises? Are your auditors beginning to pay attention to application security controls? These are the factors that will allow you to make a convincing case to management.

Are there any legal and/or regulatory requirements in relation to application and software security?

Legal and regulatory requirements vary in the extent to which they designate specific security controls. Some of them state a high-level mandate that requires organizations to protect sensitive information. Others may get more specific, actually mentioning the need to pay attention to software security. For instance, FFIEC guidelines, which apply to financial institutions, mention the need to incorporate security controls in application software. The Payment Card Industry Data Security Standard <https://www.pcisecuritystandards.org/> also addresses the need to secure applications involved in credit card transactions.

In general, if your application processes data that is covered by a law or a regulation, be sure to understand which controls you may be required to implement. It is wise to consult with a legal counsel to determine which laws and regulations apply to your organization, and the extent to which such requirements cover software security.

How do I ascertain or understand what the diverse range of software security requirements may be throughout an enterprise?

A risk assessment is a good starting point for understanding software security requirements throughout the enterprise. While conducting the assessment, you will determine which applications you need to cover, what vulnerabilities you may need to address, and what threats you should worry about. You will also learn the value that business groups assign to the applications. All this information will help determine the requirements that a software security program will need to accommodate. Some of them may apply to all business units; others will be relevant to specific groups.

Be sure to solicit feedback from both technical and business users to obtain a comprehensive perspective on the organization's needs.

The Software Development Life Cycle is based on delivering functional requirements. How can software security have any functional relevance?

Security controls, such as those related to authentication, authorization, and data privacy, can be defined as functional requirements. For example, a mechanism to validate user input will have certain inputs and outputs while performing a particular function.

Similarly, a mechanism for creating users with the appropriate privileges can be defined as a functional requirement. Other security controls may be defined as non-functional requirements to impose certain restrictions on the application's functional requirements to meet the necessary control objectives.

Keep in mind that while functional requirements are often described as "things the application must do," they are more properly viewed as the set of constraints on the application's behavior. That is, functional requirements should really be read as, "The application should do these things and only these things." In this light, software security defects can be seen as things that contravene functional requirements by allowing the application to exhibit behavior outside the implied constraints.

Section 2 – Starting Out

My organization has stipulated that software development activities associated with new projects and major upgrades to existing applications must be conducted in accordance with our new Software Security Policy and Standards. While it is easy to understand what a new application is, what qualifies as a 'major upgrade' to an existing application'?

A 'major upgrade' typically refers to a change that affects a significant part of the application's functionality, its code base, or its users. It is an ambiguous term whose interpretation could depend on the application, the organization, and even the individual. As a result, the organization should offer guidance regarding what it considers a 'major upgrade.' Even with such guidance, the term can be easily misused or misunderstood.

When an important decision hinges upon a proper interpretation of this term, consider validating your understanding with your peers and your manager.

Similarly, what would then be considered a minor release, a hot fix, and a patch?

A 'minor release' is often refers to a revision that corrects a specific issue, modifies a particular function, or introduces a small number of features.

As with a 'major upgrade,' 'minor release' is a loaded term that can be easily misunderstood or misused. The same advice applies as with the 'major upgrade.'

Section 3 – Security Risk Management

Section 3 – Security Risk Management

What is meant by the term "security risk"?

Security risk is the degree to which an application (and its data sources) is vulnerable to one or more threats.

A threat is any possible danger that an application (or more widely, a business) could be exposed to. For instance, a potential threat to an online banking site is the possibility of losses due to theft from customer accounts.

A vulnerability is any weakness that could be exploited in an application - for example the lack of strong authentication controls would make it easy for an attacker to gain unauthorized access to customer accounts and steal money.

Threats and vulnerabilities by themselves are not necessarily problems. For example, although an application may have poor access controls (the vulnerability), the threat of theft may be somewhat limited if there are no databases containing confidential personal information. It is when threats and vulnerabilities combine that we can derive the amount of risk as shown by the formula:

Risk (due to a threat) = Threat x Vulnerability (to that threat)

What is meant by the terms "risk assessment", "risk analysis" and "risk management"?

Risk Management is a collective term for activities associated with risk assessment and risk analysis.

Risk assessment activities are those that are related to determining an application's level of vulnerability to an attack.

Risk analysis is considered the process whereby cost-effective security/control measures may be selected by balancing the costs of various security/control measures against the losses that would be expected if these measures were not in place.

Risk management is the on-going process of assessing risk, taking steps to reduce risk to an acceptable level and maintaining that level of risk.

Are there any standard security risk management methodologies available?

There are a number of methodologies available such as OWASP's CLASP (Comprehensive, Lightweight Application Security Process) <http://www.owasp.org/index.php/Category:OWASP_CLASP_Project>. There is also the STRIDE/DREAD model used by Microsoft <http://www.owasp.org/index.php/Threat_Risk_Modeling#STRIDE>.

When should risk management activities be conducted and by who?

Risk management activities should be conducted in the following three circumstances:

1. When kicking-off any new development projects.
2. When performing major updates to existing projects.
3. On a periodic basis (e.g. once yearly) to account for the accumulation of small-scale (but continual) maintenance activities.

The key people who should be involved in risk management activities would include at least architects and their management, especially those with security expertise.

I am only developing a small scale application - surely I don't need to bother with risk assessment? Right?

The problem with 'small scale' is that it can mean so many different things to so many people - lines of code, number of users, number of transactions, classification of data, etc.

Although an application may have a relatively small user base, it may be dealing with a lot of highly confidential personal information. Small scale in this context is wrongly attributed to preconceived ideas of 'size' in terms of numbers as opposed to the consequences of not protecting personal information adequately.

Risk management looks to be a labor intensive and costly process. How can it positively impact the 'bottom line'?

To put the business case for risk management into context, think of the recent TJ Max case that resulted in the personal details of at least 31 million customers being stolen.

Based on the information available, it would have cost the company a lot less to have conducted a thorough series of risk assessment activities and implemented any required security controls and processes, in comparison to the cost of settling a class action suit.

The point of highlighting the TJ Max case is that risk management should not be viewed as being a short term cost center, but as a long term investment that protects the interests of a company, its shareholders and customers in the long term.

Section 4 – Security Architecture

Section 4 – Security Architecture

What is meant by a "Security Architecture"?

A Security Architecture is described in many ways. Some people consider it as a description of how a system's technical components are put together to satisfy the security requirements. Others consider it as the processes surrounding user authentication and authorization. Others concentrate on the network.

A Security Architecture is really a set of recommendations, standards, principles, and best practices addressing the people, the technology, and the operational aspects of the information system in order to address security requirements while the information system is fulfilling its purpose.

What sort of content should be included in a Security Architecture document? Is a single document sufficient?

Information security touches almost every aspect of a system and its use by people. You probably will not find all the information related to the security of a system, a network, or an enterprise in one document. However, you should be able to find at least one document that summarizes the pertinent elements of the Security Architecture. The overall document may contain references to supporting system documentation that can provide more granular detail.

The primary 'Security Architecture' document should contain, as a minimum:

- The vision and scope of the document (are you addressing the needs of a global corporation, a LAN, a supplication, or the distribution of data across several systems?).
- The purpose of the Security Architecture (are you providing public access to governmental information or restricted access to health or banking data?).
- The elements of the Security Architecture (what are the domains involved such as operating systems, networks, data and software?) and how they map to the requirements.
- What are the guiding principles (is the Architecture based on the use of open standards or .Net?)
- What are the applicable best practices (what approaches have consistently demonstrated by other organizations to achieve similar results, which in the case of Security Architecture, is demonstrating the principles?)?

Section 4 – Security Architecture

At what stage of the Software Development Life Cycle should Security Architecture documentation be created?

The IEEE P1074 Standard for Developing Project Life Cycle Processes, is one of the few - or perhaps only - project life cycle methodologies built to address security. Such a standard proposes dealing with security throughout the life cycle of the software rather than handing it off to one development group at the end, giving you a plan for including all aspects of the software development life cycle (SDLC) when making security-related decisions. It puts projects in an enterprise business context and it provides the framework for coordinating software security efforts across all disciplines and over the lifetime of the software.

Another useful document to consult for further advice is NIST Special Publication 800-64.

Whose responsibility is it to create a Security Architecture?

Computer security has become a critical business concern, and, as such, the responsibility of all IT professionals. The creation of a Security Architecture, therefore, is in actuality a collaborative process involving all stakeholders in system design and implementation.

What is a "Security Architecture Review" and what is its relevance?

A Security Architecture Review can occur at several levels. A review should be held when an organization's Security Architecture is first formulated in order to ensure that it is both compliant with security requirements and internally consistent. In this case, the review would focus on the Security Architecture being the specifications for building a system, similar to the architectural requirements used to generate the blueprint for building a house.

The next step, such as building an application, takes this specification to the actual design level. The cost and effort of retrofitting security after development are too high.

A Security Architecture and Design Review helps you validate the security-related design features of your application against your Security Architecture, before you start the development phase. Here you are making sure that the blueprints for a project meet all the necessary requirements and specifications. This second review allows you to identify and fix potential vulnerabilities before the fix requires a substantial reengineering effort.

Section 5 – Customer and Internal Applications

Section 5 – Customer and Internal Applications

We develop a range of internal applications for company employee and contractor use, as well as web based applications use by our customers. Are there any differences between the two types of applications in terms of software security requirements?

Any differences in security requirements for internal and customer facing applications will largely depend on their intended function. For instance, an internal HR application may deal with sensitive employee or contractor personal information. In this case you will need to carefully consider the use of encryption and role based access. On the other hand, if you have a basic customer web application that is used for informational purposes only, then encryption and role based access is not likely to be high on your list of security requirements.

In an internal application that uses access control, possibly extending to role based access; it may be prudent from a security point of view to use an existing LDAP directory, which could be used to provide single-sign on facilities across a wide range of internal applications. With a customer facing application, you may be implementing a database driven access control model. The principal requirement in this context is to never store user passwords 'in the clear'. Instead, use a password hashing mechanism such as SHA-1 to encrypt passwords.

While having a password policy that is as stringent as that required for employees (in terms of complexity and length, etc) on a customer facing website may seem to be the best in terms of security, it may not be the most usable for customers. In this context be prepared to implement a softer approach to password requirements in the interest of customer usability too. For instance, it may not be feasible to require customers to change passwords every 60 days. However, it would be prudent to advise customers that changing passwords on a regular basis is a good security practice.

Section 6 – Platforms, Databases and Tools

Section 6 – Platforms, Databases and Tools

Is software security more specific to any particular operating system?

In the broadest and most basic context, there are two different types of software - end user software that we produce to service some type of business requirement, and software based tools used to either build end user software or enable it to run effectively.

The latter type of software includes all the popular development tools and environments, databases and operating systems that developers use to produce the former type of software - everything from online banking applications and e-commerce websites all the way through to blogging websites (and almost everything else in between).

It can be argued that Windows based operating systems and server applications have traditionally been less secure than their Unix/Linux equivalents. For instance, Microsoft's IIS server up to (and including) version 5, has been the single most hacked web server on the Internet because of so many security flaws that have been exploited. However, in Microsoft's defense due to its regular patch release program, the ease by which end users can install patches via Windows Update and also the widespread use within Microsoft of their own Security Development Lifecycle methodology; Microsoft produced software is becoming more secure.

However, the biggest software security challenges are often to be found in how websites, end user and line of business applications are built using any of the available software tools already mentioned. If the actual application itself is flawed from a security architecture point of view, the fact that it is running on a Linux based Apache Web server that uses MySQL, does not make the application any more secure than a Windows based equivalent.

We only produce client-server applications; surely I don't need to be concerned about software/application security?

We live in an ever web-connected world which is the source of an increasing amount of software security challenges. Although a traditional client-server application may not be connected today, this may not be the case tomorrow, either by design or accident. For instance, even though a client-server application is designed to be used in a closed environment, it may still be accessed via a Virtual Private Network (VPN) connection over the Internet. In this context, a traditional client-server application needs to have strong authentication and role-based access controls in place.

As an alternative to VPN based access, if a client-server application is installed on a computer that can be accessed via the Remote Desktop Protocol (i.e. Terminal Services etc), authentication and role-based access controls are still issues that need to be addressed.

We have a lot of legacy applications that would simply cost too much to re-develop. Are there any security implications of integrating our legacy applications with web based front ends?

Given the potentially prohibitive costs and development timescales associated with completely re-developing mainframe based legacy applications, web services are seen as a key business enabler. The two principal concerns associated with such projects are access control and confidentiality.

In a legacy environment it may have been relatively easy to manage who had what type of access control to a specific application or data source. However, in a web-enabled environment, the number of potential users may increase exponentially and this must to be carefully designed and implemented to ensure the rule of least privilege is accommodate.

In a closed mainframe environment, data may have been transmitted between different locations using dedicated communication lines (i.e. leased), and therefore not directly accessible to just about anyone. Because the security risk may have been substantially lower, there would have been less need to encrypt data while in transit or stored at rest.

In a web enabled environment, instead of closed leased lines, mainframe data is now being transmitted across the open platform that is the Internet, to a much larger user base. In such an environment it is vitally important to protect information while in transit and when being stored at rest too. To protect information in transit, technologies such as Secure Sockets Layer (SSL) can be used. However, since the mainframe applications may not have been designed to encrypt information at rest, special consideration of the web-enabled security architecture needs to be focussed on the web services that will have to handle this function.

We are a predominantly Microsoft technology focussed company, but most of our projects are classic ASP based. Are our applications any less secure than .NET applications?

Although .NET may have more in-built security features such as forms based authentication and the use of compiled source code as opposed to interpreted source code, 'classic' ASP applications are not necessarily any less secure than their .NET counterparts.

The level of security within an application is more associated with how it has been designed and programmed as opposed to the platform itself. For instance, if you take a look at any of the vulnerabilities covered in the OWASP Top 10 document (http://www.owasp.org/index.php/Top_10_2007), they actually describe vulnerabilities that are associated with the use of any language used to develop web applications.

The emphasis should actually be on the design of the applications themselves to ensure that they protect against the most common types of vulnerabilities. In addition, the quality of security within an application should also address any principal security risks identified during any risk assessment activities. For instance, if an application uses highly confidential information such as Social Security Numbers (SSNs) it should be strongly encrypted using an industry standard method of encryption. If a 'classic' ASP application has incorporated such a control within its security architecture, it can be considered to be more secure than a .NET equivalent which does not use encryption to protect highly confidential information.

Our company is primarily focussed on web based applications. Is there any single development platform that is more secure than others - for example .NET versus PHP etc?

Proponents of both .NET and PHP (and indeed other platforms such as Ruby on Rails and Perl) may enthuse that their chosen development language/platform is more secure than the others. However, very often, such debate is based on the security features that are available to each language/platform, as opposed to their implementation within the context of an overall security architecture.

For further details on the importance of having a secure architecture as opposed to relying on security features, consult the "Security Architecture" section of this document.

Up to now we have been developing classic ASP applications - if we port all existing applications to .NET, will this make everything more secure?

The practice of 'porting' an application from one platform to another is based upon the premise that a new application uses the exact same architecture as the older application. A possible flaw in such an approach is that it assumes that somehow just by using newer technology, the application will somehow become more secure.

Some people may claim that since compiled code is deployed to a production server (.NET) as opposed to interpreted scripts, that their application is somehow more secure. However, such a statement is not entirely true. If uncompiled .cs or .vb 'code behind' files are used instead of assemblies - source code is still being deployed to a production server, which could provide an attacker with a lot of knowledge if they can get access to those files.

Leaving aside the arguments of compiled assemblies, interpreted scripts and uncompiled code behind files, if the original architecture is inherently flawed from a security perspective, the new application will suffer from the same problems!

If production timescales and budgets permit, always consider performing a risk assessment of any existing architecture for security purposes before it is ported to an alternative technology platform. Perhaps since the last major release of the legacy applications, there may have been various ad hoc updates performed that may have an unintended impact upon the overall security architecture. At the very least, code analysis tools should be used to detect the existence of common vulnerabilities. Ideally any new code developed should be subject to a secure code review.

Remember, if the application being ported involves credit card information, a secure code review will have to be performed anyhow, if a web application firewall is not being deployed to comply with the PCI Data Security Standard <https://www.pcisecuritystandards.org/>.

We use quite a variety of database technologies in our organization. Are there any reasons from a security point of view why we should select one over the other?

Most modern server based database technologies offer excellent levels of functionality and features relevant to security such as role based access, field level encryption etc. In addition, popular databases such as Oracle, SQL Server, MySQL and PostgreSQL all offer their own programming facilities to create stored procedures, thus helping to provide full n-tiered application solutions.

While some vendor solutions may lay claims to be more scalable than the others and offer more 'secure features', what really matters is the security of the overall database design and how the database server is configured.

From a design perspective, if end user passwords are stored in clear text, this is still poor security design, regardless of the actual database product in use. Likewise, if a database is poorly configured and uses the same account to access the database both for anonymous public internet users and administrative console access, there are a variety of vulnerabilities that could be exploited to negatively impact the confidentiality, integrity and availability of the database.

Apart from database design and configuration issues, in an enterprise class application scenario, always choose to use a server-based database solution such as the products mentioned in preference to a file based solution such as Microsoft Access. Although it is possible to use stored queries in Microsoft Access, the access control mechanism is substantially weaker, which can be easily compromised.

Section 6 – Platforms, Databases and Tools

I have a project that will need to access two different databases that are in separate locations and use different technologies. What should I be concerned about from a security perspective?

Applications are becoming increasingly complex in terms of the mix of technologies in use, evolving business practices and requirements, and also the reliance upon an infrastructure comprised of multiple components that are physically and logically separate from each other.

As part of any risk assessment activities in this type of scenario, it is vitally important to identify all data flows as they leave one system and enter another. Identify what types of data elements are associated with each data flow, in what combination, who are what initiated the transmission of information and who or what is receiving the data. Once such details have been established, it will be easier to assess your overall exposure to risk and then decide upon any mitigating security controls or processes. For instance, aside from possibly encrypting data in transit, it may be prudent to implement validation checks not only to ensure that any data is well-formed, but it has been sent from an authorized source.

Equally important in a scenario where information can flow between multiple points is the need to validate requests for data, or for transactions to be performed. In this context it is important to ensure that the initiator of the request is permitted to make the request in the first place.

Section 7 – Development Best Practice

Section 7 – Development Best Practice

Where can I learn about security best practices in relation to software development?

One of the most widely known sources of security best practices is the Open Web Application Security Project, usually known by their acronym 'OWASP'. According to their website (www.owasp.org), OWASP "*... is a worldwide free and open community focused on improving the security of application software*" whose mission is *"...to make application security visible, so that people and organizations can make informed decisions about application security risks*".

OWASP sponsors a wide variety of 'best practice' based projects, including:

- The OWASP Top Ten
  http://www.owasp.org/index.php/Top_10_2007
- The OWASP Guide Project
  http://www.owasp.org/index.php/Category:OWASP_Guide_Project
- The OWASP AppSec FAQ Project
  http://www.owasp.org/index.php/Category:OWASP_AppSec_FAQ_Project

Isn't the SANS Software Security FAQ just a repetition of the OWASP AppSec FAQ project?

No. The OWASP AppSec FAQ project is more focussed on the technical aspects of secure coding that are specific to developers. On the other hand, the SANS Software Security FAQ (this document) is focussed on wider range of security issues that are relevant to the entire software development lifecycle. In this context, both documents are complementary to one another!

If my developers follow the OWASP top ten, will this be sufficient to secure our applications?

According to OWASP, the Top Ten document "*... represents a broad consensus about what the most critical web application security flaws are*". Note the emphasis on the word 'critical'. Although implementing the Top 10 will get your 'security push' off to a great start, it is by no means an end in itself. How secure an application is will depend on the types of processes and controls that have been implemented, appropriate to that application's level of security risk.

Section 7 – Development Best Practice

What is meant by "Security Refactoring?

Software refactoring is the process of enhancing or improving the quality of code in a program. Refactoring does not alter the functionality of the software; but, it makes it more maintainable for future changes.

Security refactoring takes refactoring one step further and focuses on the implementation of security best practices such as the OWASP Top Ten. In this context, error management techniques may be added to an existing code base so as technical error details are not revealed to an end user via a browser displayed stack trace.

Is it better to refactor or simply start again?

From a security point of view, refactoring is (to borrow from management speak) concerned about 'doing things right'.

While refactoring may address certain types of vulnerabilities, it is not really concerned about 'doing the right thing in the first place'. In this context 'doing the right thing' would mean reassessing the entire security architecture of an application from a risk management point of view.

Unless there are only a small number of coding related vulnerabilities, performing a wholesale refactoring exercise could be regarded as wasted effort. Although you may have implemented server-side input validation controls, it may be pointless if the system architecture itself is fundamentally flawed from a security point of view.

Some development teams in our organization have a more mature understanding of, and approach to software security than others. Ideally what I would like is to leverage the expertise that already exists to raise the bar for everyone. However, not all teams wish to share and learn as much as each other - what should I do and how?

Every organization is subject to some degree of "office politics", that can manifest itself in the defensive attitudes of individuals and entire teams who feel threatened by quality improvement initiatives. Often what is needed in such circumstances is to have one or more 'security champions' within an organization who (skilled in the arts of diplomacy and marketing) can identify best practice within teams and promote it company wide. One way of making this approach work is to tie security best practice into the need to comply with process improvement initiatives such as Capability Maturity Model Integrated (CMMi).

What is a secure code library and what is the benefit of having one?

A secure code library is central repository for developers to store code that is designed to be re-usable on a project to project basis. The ultimate code library would contain basic functions and routines that could be used company wide by any team using a particular development language.

Code that would be suitable for a code library would include authentication and access controls, as well as white listing based input validation routines such as regular expressions to prevent cross-site scripting, etc.

The benefit of using a code library for such common functions is that there is no need to for people to continually 're-invent the wheel'. No two developers will approach the same problem in the same way. In this scenario, each developer will introduce levels of complexity that can not easily be utilized by people other than themselves (perhaps due to poor or non-existent documentation). Complexity in turn, is often a source of security risk, and therefore best avoided.

Code libraries are all very well in principle, but nobody uses the content or actively contributes to it. Isn't this a waste of time and effort?

The key to successful code libraries is continually identifying sources of quality content and then actually requiring people to use such content.

Code reviews are a great opportunity for a team to discover code that could be reused on a project to project basis. Process quality initiatives and also security policies and standards can be also be utilized to help ensure that code libraries are used, due to their compliance based design.

How can I be assured that any required technical security controls have been implemented to mitigate any known risks - before the application is deployed?

The use of the term 'required security controls' within the question could be used to infer that risk assessment activities have already been conducted. What is required in this context is to actually check that the required security controls are put in place. Checklists comprising not only of common types of vulnerabilities but also specific types of security features can be completed during code reviews and at other critical milestones throughout the software development life cycle.

Having an actual checklist and filling it out is simply not enough - developers and other project members must be required to put their names to specific checklist items to make them accountable for their work and the decisions they

make. Prior to a project being deployed, the project lead should carefully examine all applicable checklists as a final quality gate check to ensure that everything has been completed as required.

Section 8 – Source Code Review and Analysis

Section 8 – Source Code Review and Analysis

What is a Source Code Review?

A Source Code Review is an examination of an application's source code to identify any areas where it diverges from the correct, safe implementation of the application's functional requirements.

What does Source Code Analysis mean?

Source Code Analysis normally refers to the use of software tools that can automatically scan source code for evidence of any common security vulnerabilities.

What's the difference between static and dynamic code analysis tools?

Static code analysis tools analyze application code without executing it. Dynamic code analysis tools interact with a running instance of an application, and never require source code. Each type of tool has particular strengths and weaknesses.

Static code analysis tools do not require a runtime environment and so can be more convenient to use than dynamic tools. Static tools often accept source code as input, and can therefore trace problems back to locations inside the source code, making it much easier for software developers to locate, understand and fix problems. They do not require input test cases to be created and carried out, and they can analyze infrequently executed code branches as easily as frequently executed ones, which is useful in identifying problems that are difficult to reproduce at runtime. However, static tools cannot always determine more complex, data-driven execution paths, and configuration of the tool for a complex application can require significant effort.

Because they examine the actual execution of an application, dynamic code analysis tools have the potential to access execution paths that static tools cannot. The other side of this coin is that dynamic tools cannot analyze the behavior of code that does not run, but could conceivably run given the appropriate inputs to the application.

Static tools excel at answering the question, "what might possibly happen when this application runs, erring on the safe side?". For this reason, they are especially valuable for security reviews, and for detecting program defects that are difficult to reproduce at runtime. But due to the nature of computation, they have some inherent limitations. They are unable to ignore many conditions that cannot arise in practice, leading to a relatively high proportion of false positive findings, and they cannot guarantee complete coverage for programs with sufficiently complex behavior.

Dynamic tools excel at answering the question, "what specifically happens when this application runs with particular inputs?". For this reason, they are especially valuable for performance profiling, and for detecting data-driven execution paths that static tools can miss.

Should separate Source Code Reviews be developed for functionality and security?

Yes. These two types of reviews require different skill sets and are prioritized differently, in terms of depth and frequency, for the purposes of risk management.

Another reason to perform these reviews separately, even when the resources to do both are available in a single team, is that different mindsets are required for each type of review. Avoiding the need for reviewers to cognitively "switch gears" during the review process results in a more efficient use of resources.

A review focused on functionality is concerned with the question, "Does the source code correctly implement the functional requirements?". Contrastingly, a review focused on security asks, for each functional requirement, "Is there a way the implementation can be exploited to achieve some unintended effect?". Note that this latter question assumes an ability on the part of the security reviewer to determine the intended functionality. Well-commented source code, accurate documentation, skilled reviewers, and the availability of a knowledgeable development team for consultation during a review are all of invaluable assistance in making this determination.

Apart from developers who else should be present at a Source Code Review?

The review of source code is an intrinsically technical activity, and as such little benefit to the development process is likely to be gained by the involvement of non-developers. Though individuals with a deep knowledge of the application's domain and requirements might be helpful in situations where the requirements have been garbled in the translation from human language to computer language, these situations are probably most effectively dealt with during testing, not a source code review.

Are there any readily available application security checklists that may be used within Source Code Reviews?

An excellent starting point for a security checklist would be the OWASP Top ten. Microsoft also published its own series of checklists as part of its "Security

Development Lifecycle". SANS provides the "Auditing Application Development" (http://www.sans.org).

If I use code analysis tools, does this mean I don't need to do any manual source code reviews?

No, manual source code reviews are still indispensable. Although code analysis tools are invaluable in assisting with a source code review, human involvement is necessary to evaluate the relevance of the output the tools produce, and to discover problems that are at too high a conceptual level, or too subtle, for automated tools to find.

Section 9 – Testing

Section 9 - Testing

How can software be tested from a security point of view?

Too often when this question is asked, there is a dangerous implicit expectation to add security to an application via the testing process. Security as it applies to web applications should be integrated into the Software Development Cycle at all levels. Security requirements must be defined along with business requirements, threat modeling / risk analysis decisions integrated into the architecture and design of the application, in addition to security training and code reviews integrated into the implementation.

Secure code analysis is performed during the development phase. The tools used here should focus on common coding mistakes with regard to the security implications of those issues.

Software security testing should be implemented next during the testing phase as the application functionality is tested. Security testing at this level should focus on finding vulnerabilities in the web application, the web support framework (i.e. IIS, Apache etc) and any vulnerabilities at the operating system level that may impact the security of the application.

The final component should be testing of the completed web application prior to entering production. Such testing ("penetration testing") should be focused on finding vulnerabilities and then attempting to exploit those vulnerabilities to determine if they present an issue to the security of the web application and its data sources.

How can testing be used to ensure that any required security controls have been implemented to address any documented security risks?

The security testing needs to be based on defined security requirements, and the analyzed risks and chosen security controls to mitigate the unacceptable risks. Abuse cases are defined (preferably in the earlier phases of the SDLC cycles) to which test cases are then developed to ensure the security controls are effective in mitigating against analyzed risks.

What is meant by "Penetration Testing" and just how useful is it?

Penetration testing involves looking for vulnerabilities in an IT solution and then attempting to exploit those vulnerabilities to determine if they present a significant security risk to the solution. The tester, sometimes known as an ethical (or "white hat") hacker, generally uses the same methods and tools as a real attacker. Afterwards, the penetration tester will provide a written report of all vulnerabilities discovered and suggest steps that should be taken to make the system more secure.

Section 9 - Testing

In many situations, penetration testing is only slightly more than automated vulnerability scanning with some minimal manual testing and analysis. With regards to web security, penetration testing must include the web application, the web application support framework, and the underlying operating system installed on the web server. At any of these levels, issues can exist that can allow exploitation of the security and facilitate unauthorized access, even indirectly to the application.

Now, in practice there are three major problems that can make penetration testing ineffective, because its intent is to find and demonstrate exploitations of one or more serious vulnerabilities:

- It is not a systematic attempt to find all high-risk vulnerabilities.
- Because it is necessary to demonstrate the exploitability of the vulnerability, additional time and resources are unnecessarily consumed.
- It is done as a minimal knowledge black-box test, and therefore is less effective than a white box security testing which makes full use of the insider knowledge, documents and source code available.

On the positive side there are some aspects of penetration testing we need to borrow. Security testing needs to be a maliciously motivated to break-it, throw-everything-we-got-at-it, style of testing, conducted by a group independent from the designers and implementers.

Penetration testing also works very well, for example, in getting management and leadership to understand the real-world nature of weaknesses that may otherwise be considered theoretical. Penetration testing also works very well to meet the auditors' or regulators' narrow, yet common requirement that it be conducted on a periodic basis.

How do you perform a Penetration Testing?

Penetration testing must involve automated and manual tools and techniques to ensure thorough testing.

Automated testing uses tools that performs tests based on predefined databases of signatures for known vulnerabilities and attacks discovered in the wild. Some tools can take the testing to the next level by "fuzzing" or taking the standard signature and "mangling" it to see if doing so can identify a potential security vulnerability within the web application.

Automated testing can address approximately 80% of the available attack surface in a web application. It cannot account for custom code or non-standard code implementations. Developers, even when working from a predefined code base, can take the same code snippet and implement it in two different ways. For this reason, manual security testing tools and techniques should supplement automated testing in order to cover this general issue. By manually testing the web application and observing the responses, the web

application can be thoroughly tested to ensure the maximum level of security compliance.

Should developers be trained in penetration testing?

Developers do not necessarily need to be trained in penetration testing. Like any employee in an organisation, they should be given general training to gain an overall understanding of the importance of security. They should also obtain training in secure coding practices so they can better understand how minor changes to the way they write code can go a long to improving the security of the finished product. Further advice in relation to training developers is available within the "Training and Awareness" section of this document.

There are other aspects to penetration testing, such as how to exploit vulnerabilities once they are found, which are not necessary for developers.

How should software security testing be documented? For example, should security testing results be documented separately from functional testing results?

Security testing is different from functional testing, in that it focuses on how the application should not break when abused or should fail gracefully in an expected manner. In spite of this difference, it is usually better that security requirements be included right along with feature requirements, and security test cases (abuse cases) be integrated along with feature test cases (use cases). The integration is helpful because the requirements and test case are interrelated and will fit into the existing structure. For example, if a feature is added or removed, the security testing for that feature needs to be added or removed along with the functional testing. Also there will be some overlap with cases testing security as well as features.

What are the differences between sandbox/development testing, system testing and regression testing?

Sandbox/development testing is also known as unit testing and is performed by the developers in a sandbox environment, which may not include the full application, and is not as formally controlled.

System testing is done by a separate test group, with the full application in a controlled environment.

Regression testing is a test intended to be repeated, to ensure that previously working behavior does not regress (break). Regression testing is usually automated or at least semi-automated since it is expected to be re-used through multiple development cycles.

Section 9 - Testing

Can I perform sandbox/development testing on the same physical hardware as a production server?

Development testing usually occurs on a server that, by its very nature is not usually configured to the production standards as the Internet web server the application will eventually reside on. As such, the testing is not thorough and complete.

Ideally the development team creates and tests the code on a development server. They then transfer the application to a test server where sandbox, vulnerability and functionality testing occurs.

Finally, the application is transferred to a staging server which is a mirror of the production server and where pre-production acceptance and penetration testing occur. From here it is deployed to the production web server.

At any of the testing stages, problems can and will occur which could impact the confidentiality, integrity and availability of an application and it's data sources. Such problems must be identified and remediated before an application is deployed. Given the security risks, it is best to perform all testing on equipment that is physically separate from the production environment.

Should upgrades to existing applications be tested to the same extent as entirely new applications?

Yes. Any change to the application has the potential to change its security "profile". This extends to any feature change or functionality change including any fixes to address functionality issues. Since these changes can create an unintended impact upon or can be unintentionally impacted by the other components in the environment it is important that the application be tested thoroughly and completely to ensure that there are no unintended consequences from the change.

Where do the lines of responsibility begin and end for a developer and a tester when developing test plans? For example, should the developer only brief the tester on outline requirements who then develops the test plans?

The test plans are developed by the tester using security requirements, risk analysis and abuse cases defined throughout the SDLC. It is helpful for testers to be involved with design and code reviews, so that they are knowledgeable about the application.

Section 9 - Testing

Are there any automated security testing tools?

Yes. The most popular Free and Commercial Source Code analysis tools include: RATS, Splint, Fortify SCA, Prevent SQS, DevInspect, and Ounce. Web Security Scanners and proxy test tools include Paros, WebScarab, Burp Suite, Nikto, AppScan, and Accunetix.

Is there a standard way of classifying security defects?

Yes, several. There is the well known Common Vulnerability and Exposures (CVE) <http://cve.mitre.org/> and the Common Software Weakness (CWE) types <http://cwe.mitre.org/>. There is also the STRIDE/DREAD model used by Microsoft <http://www.owasp.org/index.php/Threat_Risk_Modeling#STRIDE>.

Is there a standard scoring system available to help me determine if an application is suitable for release - from a security point of view?

The above classification may also help in scoring, but the risk for a specific vulnerability are individual to each application and acceptable risk levels for your organization, so the scoring needs to be tailored by your risk analysis.

Are there any legal requirements surrounding the use of production data that may contain personal information, for testing purposes?

The Payment Card Industry Data Security Standard <https://www.pcisecuritystandards.org/> prohibits the use of production payment card information on test and development systems, and requires test and development system to be done on separate hardware with separate staff.

Also for the sake of protecting private information for HIPAA, privacy breach laws, and other regulations, private production information should not be used on test or development systems; or on the computers of individuals either.

Ultimately, using production data for testing purposes exposes data to risks that may threaten an organization's ability to meet the appropriate legal requirements for protecting data.

Section 10 – Deployment

What is a Web Application Firewall and should I protect my applications with such technology?

Web Application Firewalls are often called 'Deep Packet Inspection Firewalls' because they look at every request and response within the HTTP/HTTPS/SOAP/XML-RPC/Web Service layers. Some Web Application Firewalls look for certain 'attack signatures' to try to identify a specific attack that an intruder may be sending, while others look for abnormal behavior that doesn't fit the websites normal traffic patterns. Web Application Firewalls can be either software, or hardware appliance based, and are installed in front of a web server in an effort to try and shield it from incoming attacks.

Many organizations whose business involves accepting credit card payments are considering the implementation a Web Application Firewall in order to be compliant with requirement 6.6 of Payment Card Industry (PCI) Data Security Standard <https://www.pcisecuritystandards.org/>.

Installing an application firewall is just one of the alternatives - organizations can also choose to perform code reviews to help ensure that web applications are protected against known attacks. Advocates of web application firewalls will point to the speed and ease by which such technology can be put to use.

Vendors will sometimes point to the perceived costs savings attributed to web application firewalls, in comparison to code reviews which can be labor intensive and potentially more expensive. Although web applications do offer a potential 'quick fix' solution, there are a number of things to consider:

- Unless your risk and threat modelling activities revealed that you are dealing with sensitive personal information or proprietary business information, and therefore have a potentially high level of security risk, it may not be financially prudent to implement an application firewall.
- If your web applications have a high level of security risk, an application firewall is not the only answer.

Remember that you can also perform code reviews to help ensure that your applications are designed to be protected from security risks and specific vulnerabilities.

Technology solutions such as web application firewalls should be seen as providing a layer of protection within a 'Defense in Depth' security strategy. In this context if an application is protected by a web application firewall that may be compromised by an attacker, knowing that your application code has been thoroughly reviewed for security purposes will give you an extra level of protection.

If you do decide to purchase a web application firewall, you should consider a range of vendor alternatives first. An excellent source of help in this respect is to view the presentation slides from the SANS security Webcast: "*Using Application Firewalls to Comply with the PCI 1.1 Data Security Standard*".

Further information is available at www.sans.org/webcasts and selecting the 'Webcast Archive' link.

When deploying or re-deploying an application, is there anything that should be documented from a security point of view?

In short, you should document everything you do when deploying or re-deploying an application, whether security related or not. From a functional point of view, deployment documentation is essential to understand why an application has been configured in a specific way.

From a security point view, it is essential that how an application or server has been configured, is documented adequately, because any future modifications will need such detail to ensure that existing security measures are not compromised or disabled.

Apart from documenting everything detailed in this section (10), specific types of detail to document would include, but is not limited to:

- List of all new files transferred and existing files that have been replaced.
- Modifications to existing configuration files.
- Other operations performed on the server including, but not limited to, log creation or deletion.
- List of services that have been stopped or restarted.
- Details of any observations made - possibly from manual log file inspections. Such detail may be useful in the event of any incident handling procedures.

All of the above should be documented with details of who performed any of the actions, and when. Any such documentation should be stored in a central location that can be accessed by other parties that are associated with an application.

Section 11 – Operation and Maintenance

Once an application is deployed, our security worries are over - right?

Unfortunately, no! Security is an ever moving target, and new vulnerabilities will always be discovered in software. What can be regarded as 'secure' today must come with the caveat 'only for known and assessed security risks'. If you work on such a basis, you will have to assume that any software you produce cannot be 100% free of security related vulnerabilities.

More importantly though, it is essential that an organisation adopts a proactive approach to finding and remediating vulnerabilities in any applications they own and support. Regardless of whether an error is found 'by mistake' (e.g. as a result of some design flaw that an end-user 'stumbled upon' by using specific features and data), or intentionally via unauthorized use of an application, you need to be able to find those vulnerabilities and fix them.

As part of any upgrade and maintenance activities, sometimes by adding in new features, removing or updating existing features, it may be possible to introduce vulnerabilities to an application. Maintenance and upgrade activities should still place a heavy reliance upon risk assessment, security architecture reviews, manual code reviews, testing - i.e. all the accepted security best practices while developing entirely new applications.

The degree to which tasks are performed, while not being as intense or as comprehensive as new software deployments, still need to be done in some appropriate manner.

What can be done to help identify vulnerabilities in applications and bring them to our attention in the quickest and most efficient way possible?

The earlier vulnerabilities in applications are discovered the better. The best way to detect vulnerabilities in applications is actually before a line of code is produced; by thoroughly reviewing all available security architecture. When code has been written, manual code reviews and code analysis tools used appropriately can identify most of the common types of vulnerabilities that can exist in code.

Once an application has been deployed, audit logging is the best source of information to help you discover vulnerabilities. For instance, with any error management facilities built-in to applications, one of the principal objectives is to prevent technical error details from being accessible to an end user via a browser displayed trace stack. However, error management facilities should also write audit log entries for any detail that pertains to a general application design flaw or a vulnerability that is being exploited by an attacker.

Section 11 – Operation and Maintenance

What is an Operational Security Guidance Manual and why should I develop one?

An Operational Security Guidance Manual is a document that describes a variety of detail necessary to provide a sustained level of support for an application over its deployed life. There are several reasons why such a document is needed:

- Prior to the launch of a new application, the data center will need to be provided with minimum hardware specifications so that equipment can be ordered and be prepared for use.

- An application may not necessarily be installed by a member of the development team, in which case the installation engineer needs to be made aware of software configuration requirements such as services and registry entries etc. Likewise, over the lifetime of an application, it may need to be re-deployed to new hardware, in which case having access to installation, configuration and migration detail contained in an Operational Security Guidance Manual is essential.

- When performing upgrades and maintenance releases: It is highly unlikely that the original development team will always be responsible for all upgrade and maintenance activities. An operational security guidance manual is the one constant resource that should always be made available to minimize any possibility of introducing security related vulnerabilities to an application within its lifetime.

Given the accepted fact that an application will grow and change over its lifetime, always remember to keep the operational security guidance manual up to date to reflect the current security architecture.

What sort of detail should be contained in an Operational Security Guidance Manual?

The type of detail included in such a document could include:

- Pre-install Configuration Requirements: details of minimum system requirements, third party software requirements and also any infrastructure requirements including, but not limited to, perimeter protection.
- Data Resource Requirements: details of any resources used by an application, including, but not limited to, databases, configuration files, cryptographic key stores, registry keys, access control lists, and audit logs.

- Application Security Architecture: overview detail including, but not limited to, authentication mechanisms and policies for authentication, documents showing the flow of data between all application components, as well as the use of encryption.
- Security Configuration Mechanisms: Details of any configuration options required to operate the application in a manner that fulfils the system security requirements. Specific information may include, but is not limited to, web server application settings, firewall rule sets, and audit log field requirements.
- Issues: List any 'known issues and compensating controls'.

Is there any best practice guidance available on how to configure application audit logging?

Requirement ten of the PCI Data Security Standards <https://www.pcisecuritystandards.org/> provides a considerable amount of detail on what needs to be implemented to be compliant, including, but not limited to:

- Recording all actions taken by any individual with root or administrative privileges
- Use of identification and authentication mechanisms
- Creation and deletion of system-level objects.

The architecture of an application may have a variety of security risks, but not all of them will be critical. In this context you need to ensure that audit logs are not configured to record what could be considered erroneous activity, otherwise log files will grow quite quickly, and become more difficult to analyze.

Apart from storage requirements, is the need to actually review log files, often manually. Large log files containing a lot of erroneous data will therefore make it more difficult to find and assess the actual log entries that really matter. In summary, audit logging should be based on core security risks as well as available resources to process log data.

How often should application logs be reviewed?

How often an application audit log is reviewed will depend upon a variety of factors including, but not limited to:

- Types of data that is being recorded.
- The number of people using an application.
- The types of operations and transactions performed.

- Available resources (both human and technical).
- Any applicable laws and regulations.

With a customer facing web application involving highly confidential personal information, there may a greater need to review audit logs more frequently in comparison to an internal company application that uses publicly available information.

Are there different ways to classify application log entries from a security point of view?

Possible ways to classify application log entries include, but are not limited to:

- use of protocols: encrypted or unencrypted web traffic.
- data classification categories: public, internal use only, confidential, highly confidential or restricted.
- types of operation: update, edit, delete, add etc.
- role based access: use of privileged access.
- general application errors: management of syntax errors, which could include evidence pertaining to injection attacks etc.

In what circumstances should application logs be integrated with Network Based Intrusion Detection Mechanisms?

Depending on the level of risk associated with an application, it may not be enough to just create log entries.

If an audit log entry is based on application errors and highly confidential information such as a Social Security Number, an attacker may have gained unauthorized access, completed an attack and exited from the system long before the problem is discovered later. In this case, 'discovery' is by either manually reviewing a log file or when a bank receives a call from an irate customer whose funds have just been depleted. Such a scenario is an example perhaps of when audit logs should be linked to intrusion detection mechanisms so that immediate action can be taken.

One possible type of immediate action would be for the Network Based Intrusion Detection mechanism to raise an alert to a management console that is patrolled by a person. Based on the available evidence, the operator will be able to take corrective action - e.g. disable or freeze an account etc.

Section 12 – Vended Software

Section 12 – Vended Software

We are intending to use a piece of vended software 'off-the-shelf' - how should we evaluate it from a security point of view?

Although how you need to evaluate prospective vendor solutions will depend on specific circumstances such as how they will be used in conjunction with other software solutions within your company, the following are a good starting point:

- Take a look at independent product reviews for unbiased advice. For instance, the SANS 'What Works' program offers such information, much of it contributed by actual product users.
- Review any available security documentation from the vendor. Look specifically for detail that may conflict with any expressed security requirements.
- Set-up a dedicated pilot testing environment to suit your requirements. Always remember to never use production servers for pilot testing - you could be exposing your infrastructure, information, and customers to security risks!
- Set up an internal review process to ensure that prospective vendor solutions not only meet all functional requirements but adequately address any known project security risks. Such internal review processes must be based on the consensus of a number of key stakeholders. Remember to think laterally when identifying stakeholders. For instance, does a prospective vendor solution have specific infrastructure requirements for firewalls and intrusion detection? If the answer is yes, source the availability of key personnel in each area and get them on board!

We are using a vended software solution that is also hosted by the client - do we have the right to Pen Test it?

Unless you have written permission, you ***don't*** have the automatic right to pen test an application you use, that is hosted by the client. The need to pen test an application should always be included within some form of written contract or agreement.

What sort of documentation should a vended software provider be expected to make available to a customer?

Only in very special circumstances, can the prospective vendor be expected to provide access to source code for review. However, it would be feasible to expect the vendor to provide architectural overview detail, if not the actual functional and technical design specifications. If any security related detail has been independently audited, you should at least be provided with a good

overview of the detail. Depending on specific circumstances, vendors may require customers to sign confidentiality and/or non-disclosure agreements. In such a scenario, always seek the advice of your Legal Counsel!

We are intending to customize in-house, a piece of vendor supplied software - how should we approach this from a security point of view?

The most practical way of dealing with in-house customization projects is to treat them as the same as any other software development project and follow the advice offered in all areas of this document. However, there are two key additional areas that need to be focussed upon.

Firstly, is the need to identify how any in-house customized applications interface with other existing applications within your organization. More detailed risk assessments may be needed to identify any additionally required security controls.

Secondly, if the vendor has provided access to source code, they may have done so according to specifically documented conditions. In this respect, access control may be an issue whereby only specific developers should be provided access to the source code as well as supporting security related documentation.

How should responsibilities for vulnerability and patch management activities be shared between us and a vendor whose software we customize in-house?

The most practical way of managing patch management activities is to work on a partnership basis. Although the 'owner' of any specific application components will be responsible for developing the patches and possibly deploying them too, co-operation is needed in the following areas:

- Vulnerability assessment: A vulnerability in any in-house customized code may have a knock-on effect to the security of code produced by the vendor (and vice-versa). Although both parties may have their own dedicated test environments, any solution proposals need to be reviewed and agreed so as any further security related risks can be identified and remediated.
- Testing: Each party should have access to dedicated facilities to test any patches before they are deployed to production servers.
- Change Windows: While patches must be deployed in a timely fashion, neither party should perform any such duties without warning or advance planning. When a patch has been developed, tested and

certified to be ready for deployment, mutually agreed 'change windows' will help both parties manage any 'downtime'.

Central to all of the above is fluid communication in written format to minimize any misinterpretations by either party. In some cases, specific roles and responsibilities may need to be documented in a service level agreement. As with any such document, always seek advice from your Legal Counsel before signing up to anything.

We are going to have a piece of vendor supplied software customized by the vendor. What are our responsibilities from a security point of view? What are the vendor's responsibilities from a security point of view?

The vendor will be expected to deliver according to any requirements expressed within a contract and supporting documents such as functional specifications and technical design specifications. Any such documents must highlight all key deliverable requirements, including security. The onus is therefore on the customer to provide the vendor with accurate requirements, which from a security point of view, will require an emphasis on risk assessment activities.

The customer of course will need proof that not only have the deliverables been produced, but also to required levels of quality. In this scenario, any agreements and contracts must specify the customer's rights and limitations in relation to conducting penetration tests, performing code reviews and any other activities that could be termed 'customer acceptance testing'.

Our company is thinking of outsourcing some development activities. How should the prospective suppliers be able to prove they have expertise in secure application development?

There are five key sources of information that any potential outsourcing provider should be ready and willing to make available to customers:

1. Their track record: Not only case studies but also contact details of other organizations who have actually bought and use their products.
2. Resumes of key members of staff who work for the outsourcing provider.
3. Documented evidence of any quality certifications (such as ISO 27001) that the outsourcing provider claims to have.
4. Documented evidence pertaining to any application security training received by the employees of the outsourcing provider.
5. Documented evidence of any security related certifications held by outsourcing provider employees. Remember, if anybody claims to have

a GIAC certification, you can always check it out by doing a search at www.giac.org.

Section 13 – Open Source Software

Is Open Source Software more secure because we can download the source code and compile it ourselves?

Although there is a huge ongoing debate over whether or not Open Source Software is more secure than its commercial, close source counterpart, the belief that software is more secure because it is available in source code format that you compile yourself, is inherently flawed.

The process of compiling source code yourself doesn't add some level of extra security in comparison to using a pre-compiled binary. The quality of security associated with an application is in the quality of the source code itself. Unless source code provides security as a core functional component to address a wide range of identified risks, and has used security best practices to mitigate against those risks; it is no more secure than its pre-compiled binary equivalent.

The only way to know if source code provides some definable quality of security is to perform a source code review. Even if you have access to source code review documentation, unless this is available from a trusted source, it should be independently checked. For further advice on code reviews, consult the 'Source Code Review and Analysis' section of this document.

Regardless of whether you compile an application from source code or use a pre-compiled binary, you need to understand what different types of security risks the software has been designed to address and how. If you have access to security architecture information pertaining to source or pre-compiled code, read it and distribute it to other key people on your project team. Based on the information available, you can then make a decision by consensus as to whether the software meets your requirements from a security architecture point of view.

Section 14 – Training and Awareness

Section 14 – Training and Awareness

My developers are highly skilled and qualified people already, they don't need any training relating to application/software security.

Developers are not traditionally trained in the security aspects of development. Developing an application and developing an application securely are two different skill sets. Even the best developers should receive security training, and remain updated on what the current threats are, since the security field is constantly moving.

Do managers and other people in positions needed to be educated about security? If so, how?

Absolutely! Management provides the resources (such as people and money) to address the security issues. If management does not understand security, resources may not be allocated appropriately, or may not be available at all.

When security is done well by developers, it will take developers more time upfront to address the security issues, although the total amount of development time during the life of the application will go down since the same "fixes" do not have to be made at a later point in time when security issues are uncovered.

Security awareness programs can be used to educate management on some of the security issues by focusing a session specific to management. Additionally, courses exist that give management an overview of the security landscape and what threats should be considered. See http://www.sans.org/training/courses.php#management for further detail.

Are there any suitable security training courses for developers available?

Security is more of an art than a science. To that end, SANS does have courses that train developers on the security risks, so they can accurately address the risks once they are understood. SANS has a selection of courses available for developers that can be found at http://www.sans.org/training/courses.php#developer.

I recently received a resume for an open position in the development team. The applicant claimed to have expertise in secure application development - how can I verify such claims.

The best way to verify such claims is to have an experienced security minded applications developer conduct a code review of the applicant's code. Since this option is not always available, the Secure Programming Skills Assessment also exists. See http://www.sans-ssi.org/#spsa for further details.

Section 15 – Roles and Responsibilities

Section 15 – Roles and Responsibilities

There seems to be many different types of roles and responsibilities associated with Secure Software Development. How do I effectively delegate and manage such responsibilities in a range of teams that have varying resources?

The need for specific security roles and responsibilities will be dependant on a number of factors including, but not limited to:

- Project size, scope and complexity
- Timescales
- Team size
- Team skills

Because of the above factors, it may not be possible to have team members performing dedicated roles. Instead, development team members may have multiple roles - for instance a lead developer may also be responsible for conducting all code reviews. Regardless of available resources, the key to effective delegation is based on process and communication.

CMMi is firmly based upon repeatable processes that are well documented and this helps to enforce strong discipline within a team. Coupled with effective communication at every stage of the software development lifecycle, it should make the job of a project manager that much each easier.

In terms of communication, functional and technical design documentation should highlight specific security requirements. However, always ensure that written communication is backed up with appropriate briefings to remove any uncertainty in terms of what is being required of whom.

What are the key roles and responsibilities associated with Secure Software Development?

Key security roles and responsibilities include, but are not limited to:

- Security Architects - those who are specialists in risk management and can both create and evaluate secure architectures. Some security architects may also be specialists in cryptographic design and evaluation too.
- Database Administrators and Designers with responsibility to ensure that not only are database designs secure, but also are able to ensure that access control and database configuration issues are adequately addressed.
- Penetration Testers: The 'ethical hackers' who (with written permission) will try to compromise an application for the purpose of finding every possible known vulnerability.

Furthermore, it could easily be argued that if security is at the core of every stage of the software development lifecycle, then everybody including project managers, team leaders, developers, testers, as well as build and release management and co-ordinators, should each have some security roles and responsibilities.

Section 16 – Policy and Compliancy

Section 16 – Policy and Compliancy

What is a Security Policy and how does it differ from a Security Standard? What are the benefits of having such documents?

A policy is typically a document that outlines specific requirements or rules that must be met. In the software security realm, policies are usually point-specific, covering a single area. For example, an "Acceptable Use" policy would cover the rules and regulations for appropriate use of the computing facilities.

A standard is typically collections of system-specific or procedural-specific requirements that must be met by everyone. For example, you might have a standard that describes how to harden a Windows XP workstation for placement on an external (DMZ) network. People must follow this standard exactly if they wish to install a Windows XP workstation on an external network segment.

Software Security Policies and standards, if followed, are invaluable for assuring consistency in development and in use. If designed and implemented correctly with the full support of management and leadership, people will view compliancy as being part of their job role.

How do I get people to comply with a Software Security Policy or Standard?

There are two tenants to remember about good policy development: First, write your policies so that they are both operational AND enforceable. Second, good security policies should be integrated with your normal business operations to be effective. You may need to establish sanctions but, if these two goals are met, you will find that you may not need to enforce them.

Is developing a single policy or standard enough? What else will my organization need?

Developing a single policy may not be enough. You can guess at what else your organization needs or you can be more systematic. A risk analysis, formal or informal, is essential (if not mandated by regulation) to help you target the areas of greatest concern and prioritize the development of key policies. Use the risk analysis to identify those areas that need specific attention, such as the use of mobile devices (including cell phones and flash drives), determine whether administrative controls, including policy, are a part of the solutions to address each of those areas, and, subsequently, formulate and rank the most immediate needs.

Section 16 – Policy and Compliancy

What does a Software Security Policy look like? What type of information should it contain?

A standard or policy should provide a general statement that sets the goals, values, and direction of the organization, and drives the procedures, or how the organization will operationally implement those goals and values. Content normally covers the following:

- Title and version number;
- Purpose statement — the why, the goals and objectives;
- Definitions as needed to fully understand the content;
- Statement — the what, the organization's position;
- Body — text as needed to explain/support the statement;
- Responsibilities assigned to specific organizational positions to carry out the tasks associated with the policy or enforcement of the standard;
- Appendices and references — as needed to fully explain or provide additional information to carry out the policy or comply with the standard.

How do I go about developing a Software Security Policy or Standard? How long will it take?

You may have only one software security policy or you may have several. From your risk analysis, you have established an initial priority as to what policies you should tackle first. You also need to consider the present level of risk to your organization.

Remember, many security policies often build off one another, so if there is little or no security policy presently in effect, or if policy development is a new activity for most of the members of your team, you should consider starting with the simplest policies to allow the team to develop a better understanding of policy development before they tackle more complex topics. Beginning with the simplest policy allows you to and work your way up through to the more complex and challenging ones as the committee's skills and knowledge grows.

Complex policies can get caught up in the review process for a lengthy period and your team could lose momentum. It makes sense to run a few policy developments simultaneously; the high risk and simplest in parallel, but don't over-whelm the security development process. Be patient - a good policy can take six months or more to get approval.

Section 16 – Policy and Compliancy

Where can I find out more detail about the process of creating a Software Security Policy?

Although it deals more with policy development in general as opposed to software security, there is an excellent paper available in the SANS Reading Room called "Information Security Policy – A Development Guide for Large and Small Companies"
<http://www.sans.org/reading_room/whitepapers/policyissues/1331.php?portal=eb7c09f4665b3d26977811dd68ab3cbe>

How do I ensure that any Software Security Policy I develop meets the need of the enterprise?

One way to ensure acceptance is to form a Security Policy Committee who is chartered to ensure that policies meet the needs of the organization, to facilitate smooth adoption by the organization as a whole, and to determine whether a proposed policy can realistically be enforced and managed. Membership of the committee should include representatives from different business units and key departments, such as legal and human resources. Business unit staff typically understand better than anyone, what the value of the data is, what needs to be protected, and to what degree. Include representatives from whatever group performs audits within your organization. Ensure there is at least one IT representative on the committee, empowered to speak to security issues, since IT typically has the lead role in implementing security policy, especially those that involve technologies and infrastructure changes in addition to new and/or revised processes and procedures.

We developed our Software Security Policy a couple of years ago, we're done! Right?

Wrong! What good is a policy if you don't take the time to measure its effectiveness? A yearly policy review process should consider how effectively the policies align with the business objectives.

Security policies are not a write once and forget; they are living documents that will need to be reviewed periodically. The frequency will depend upon the nature of the policy and the legal and environmental changes.

Also, revisit the risk analysis that was conducted at the beginning of the policy development phase, then measure how well the policy has prevented, corrected and detected the threats to mitigate the risk to the company's data.

Section 16 – Policy and Compliancy

We've been receiving a lot of negative ad-hoc feedback to our existing policy that was developed some time ago. Much of the feedback actually seems to be conflicting too! How do I resolve the situation?

Sounds like it's time for a change. Your policies and standards may be outdated and old. They aren't conforming to your business processes or new applications. Go back and measure the effectiveness of the policy. See if the issue is understanding or whether the policy really needs to be updated. Consider additional training or awareness on the current policy or see what it will take to start the revision process now!

Last Updated: May 22nd, 2018

Upcoming SANS App Sec Training

Community SANS Pasadena DEV522 @ JPL	Pasadena, CA	May 30, 2018 - Jun 06, 2018	Community SANS
SANS Oslo June 2018	Oslo, Norway	Jun 18, 2018 - Jun 23, 2018	Live Event
SANSFIRE 2018	Washington, DC	Jul 14, 2018 - Jul 21, 2018	Live Event
SANS Northern Virginia- Alexandria 2018	Alexandria, VA	Aug 13, 2018 - Aug 18, 2018	Live Event
SANS Virginia Beach 2018	Virginia Beach, VA	Aug 20, 2018 - Aug 31, 2018	Live Event
SANS Amsterdam September 2018	Amsterdam, Netherlands	Sep 03, 2018 - Sep 08, 2018	Live Event
SANS London September 2018	London, United Kingdom	Sep 17, 2018 - Sep 22, 2018	Live Event
Community SANS New York DEV522	New York, NY	Sep 17, 2018 - Sep 22, 2018	Community SANS
SANS Network Security 2018	Las Vegas, NV	Sep 23, 2018 - Sep 30, 2018	Live Event
SANS Brussels October 2018	Brussels, Belgium	Oct 08, 2018 - Oct 13, 2018	Live Event
SANS October Singapore 2018	Singapore, Singapore	Oct 15, 2018 - Oct 28, 2018	Live Event
Secure DevOps Summit & Training 2018	Denver, CO	Oct 22, 2018 - Oct 29, 2018	Live Event
SANS Gulf Region 2018	Dubai, United Arab Emirates	Nov 03, 2018 - Nov 15, 2018	Live Event
SANS London November 2018	London, United Kingdom	Nov 05, 2018 - Nov 10, 2018	Live Event
SANS Santa Monica 2018	Santa Monica, CA	Dec 03, 2018 - Dec 08, 2018	Live Event
Community SANS New York DEV540	New York, NY	Dec 10, 2018 - Dec 14, 2018	Community SANS
SANS OnDemand	Online	Anytime	Self Paced
SANS SelfStudy	Books & MP3s Only	Anytime	Self Paced