Secure Coding.
Practical steps to defend your web apps.

Software Engineering – Security as a Process in the SDLC

SANS GSEC GOLD CERTIFICATION - 785787

Author: NITHIN HARIDAS, nithinharidas@gmail.com

Adviser: James Purcell

Accepted: April 2 2007

Outline

1. Introduction

There has been a tremendous change the way organizations tend to align with the use of Information Technology in their business. The evolving e-business trend has taken enterprise to its competitive edge. Business Applications have been evolving at a rapid pace to serve business needs. Applications are those that execute on distributed environment or any real time applications (mostly Web Based or any Client Server applications). The customer's dream of anytime access from anywhere has come into reality. In the cutting edge of the competitive world, this dream has enabled to come up with the evolution of easy to access online/real time applications with multiple functionalities to serve the requirements

Due to this rising trend in use of these applications, management tends to procure or develop applications at a faster rate. Application architects and developers are more aligned to the functionality of the applications developed. Management is satisfied since most of the applications serve and support their business needs. Hence the application development process is also aligned to the functionality and as always, security-need on these applications has come down in the priority list.

Most of the Application developers align to the Software Engineering Principles that follow through a standardized SDLC phases, but never consider or have a disciplined process to address the factor called Security in any of the phases. Does authentication and authorization mechanism (like Login and Password) on applications make them secure? Do these security considerations on developed application help them to address security in its entirety? Security attacks at the

application layer have made the organizations realize the fact that security needs to be considered at the same priority as its functionality. This paper explains about how Security as a process can be incorporated or identified in the Software Engineering principles[1] (SDLC phases) and how Organizations can leverage upon considering Security as an effective process within the existing development framework

2. The reality

There are instances at regular intervals regarding identity theft, stolen data etc. Most of the organizations are still vulnerable to a greater extent. Organizations assume that Firewall/IDS or any such Network Security devices will suffice to their security needs on the application infrastructure. Information data processed by applications are made secure by these devices. Network Security devices are more sophisticated and effective that their IT infrastructure behind those devices is protected to a greater extent. But the fact being most of the targeted attacks on insecurely developed applications will evade all these security devices and currently most of attacks are targeted at the application level.

Currently regulatory standards and requirements have established/identified the need for application security. Organizations need to think whether application security has to be considered at the pre or post development phase or throughout the SDLC process of application development?

For this reason, the developers and the architects of applications need to know the importance of securing their application and how security as a process should be incorporated during their development lifecycle.

3. Evolution of SDLC

In the past, programmers tend to develop software applications without following any engineering principles. Their core objective of the development process is to reach out the functionality aspect of the application i.e. the requirement. Most of the time, it is through experience and through various trial and error methods the application achieves its objective

This was not a success as they identified the need for a process based approach in managing large developmental projects. There arise the concepts of Software Engineering[1] Principle. The main objective of such principle is to come up with the following

- Well formulated requirement specifications (both functional and non-functional requirements)
- Well oriented - Software development life cycle (Stages that include- Development, Usage and Maintenance)
- Well defined Software development models (Engineering models)

Most of the applications were developed following a development process that is easily customizable to their management needs. Organizations were unaware of the fact that security issues those were not addressed properly during all these processes are going to hit them back with a much greater impact.

There are different standards and models for software development. None of them are growing up as a universal standard yet. Organizations as per their need

and requirement tune this process. This should be in alignment with their management process, their Quality standards mapping technology level and the business needs (functionality part)

Process improvement model like, Capability Maturity Model (CMM), developed by Software Engineering Institute (SEI)[2] was evolved to assist Organizations to ensure that their custom build process is meeting their objectives and are well tuned into their development activities.

Development Process implied in traditional software organizations has focused on

- Project Management over initial development schedule
- Managing the development costs
- Providing initial functionality to Users

Hence key objectives of Software risks (mainly Project Management risks) in place are

- Tightening budgets ensuring code re-use efforts
- Decisions on requirement analysis made early stages (development) to cut short the effort on updating code and re-engineering it

Most of the time (say 80%) spent on Software goes to maintenance cost[3]. Insufficient knowledge or awareness on the need of Security at the Software development/engineering phases may hit back. For e.g. if security has been breached on a developed software the whole saga of updating design code and even re-engineering may take place thus impacting the overall resource and cost

Most of the Organizations have Security Management process to meet regulatory and compliance requirement. As per Computer Economics[4,]

Managing risk is an increasingly important part of the job of CIOs and IT executives. Risk management includes securing corporate systems, networks, and data, ensuring availability of systems and services, planning for disaster recovery and business continuity, complying with government regulations and license agreements, and protecting the organization against an increasing array of threats such as viruses, worms, spyware, and other forms of malware. This should also include Business Critical Applications

Applications are custom built for serving or enhancing business requirement. The latest trend in financial institutions shows that the web technology has evolved to such an extent that enabled its customers with more and more functionalities. It has reflected in the areas of B2B and B2C businesses.

Organizations started developing applications to meet their customer's requirements and initiated to create a value proposition for their clients. Only objective was to make their services or products alluring to customers and at a more intimate level. Besides the Managed Security services components seen on a typical Security Management domain of an Organization, there arise the need of identifying a well defined Secure Application Development process due to the rising trend in the attacks on application level.

4. Incorporating "Security" in different SDLC phases

An evolving solution that should be considered throughout SDLC phases is as

shown in figure below. This should be carried along with the other activities ideally followed in SDLC that is not captured in this figure.

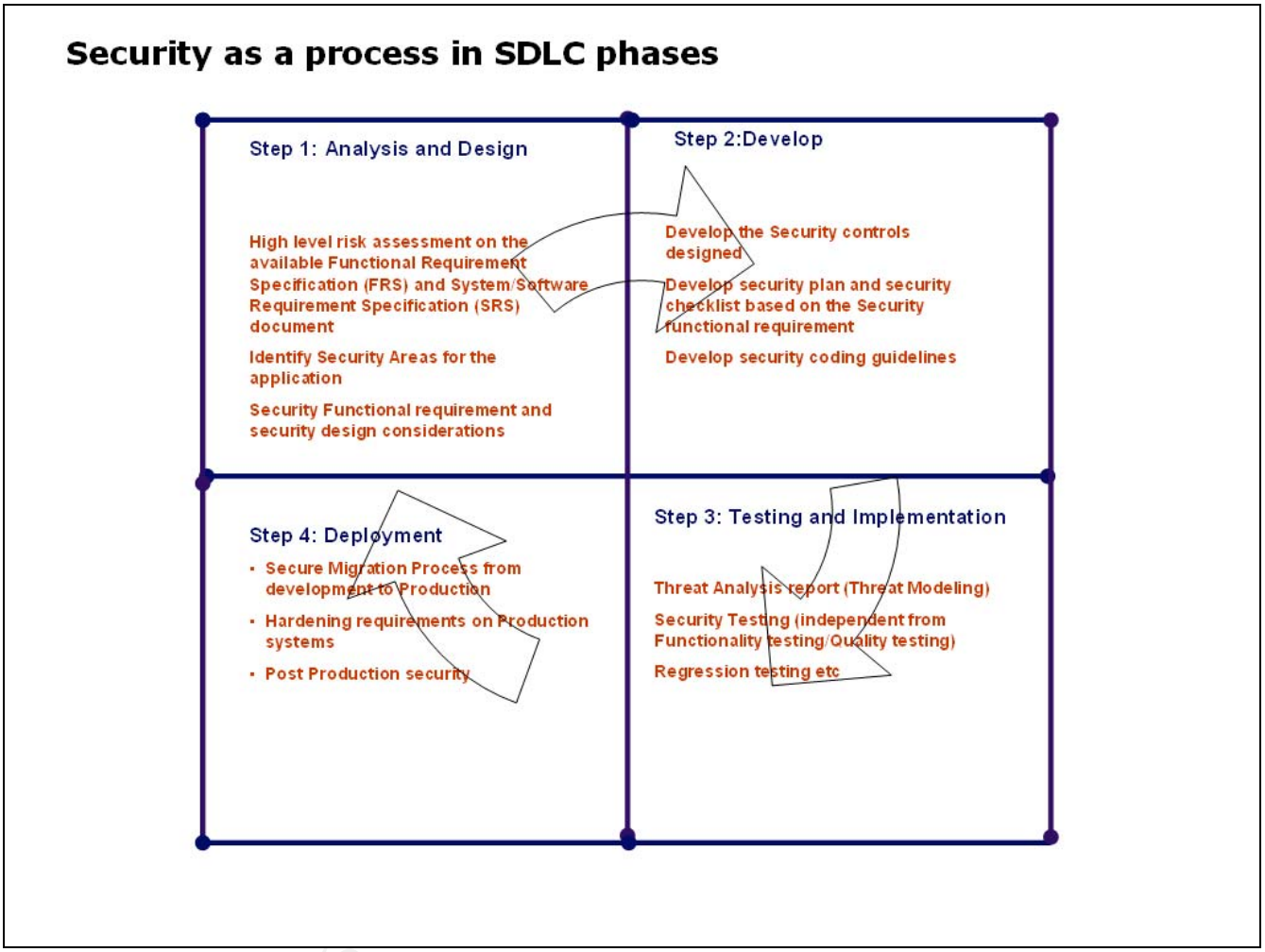

Figure 1 Security in SDLC Phases

5. Quality Framework and Security Component within the Process

Software risks impacting Quality are currently being addressed based on the existence of Quality frameworks in place. But is there any well defined and

structured framework to address Software risk at the Security level?

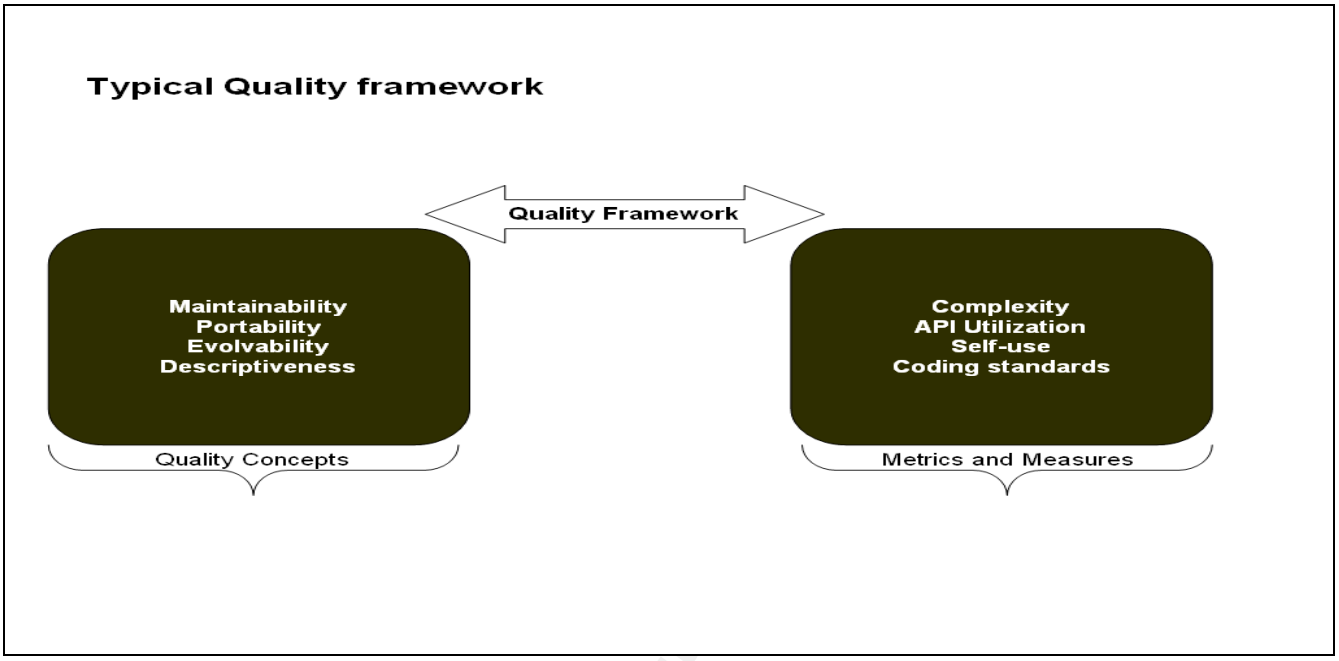

Figure 2 Existing Quality Framework[3]

Figure below tries to incorporate the Security components within an application that could be addressed in the existing Quality assessment framework (Source MITRE)

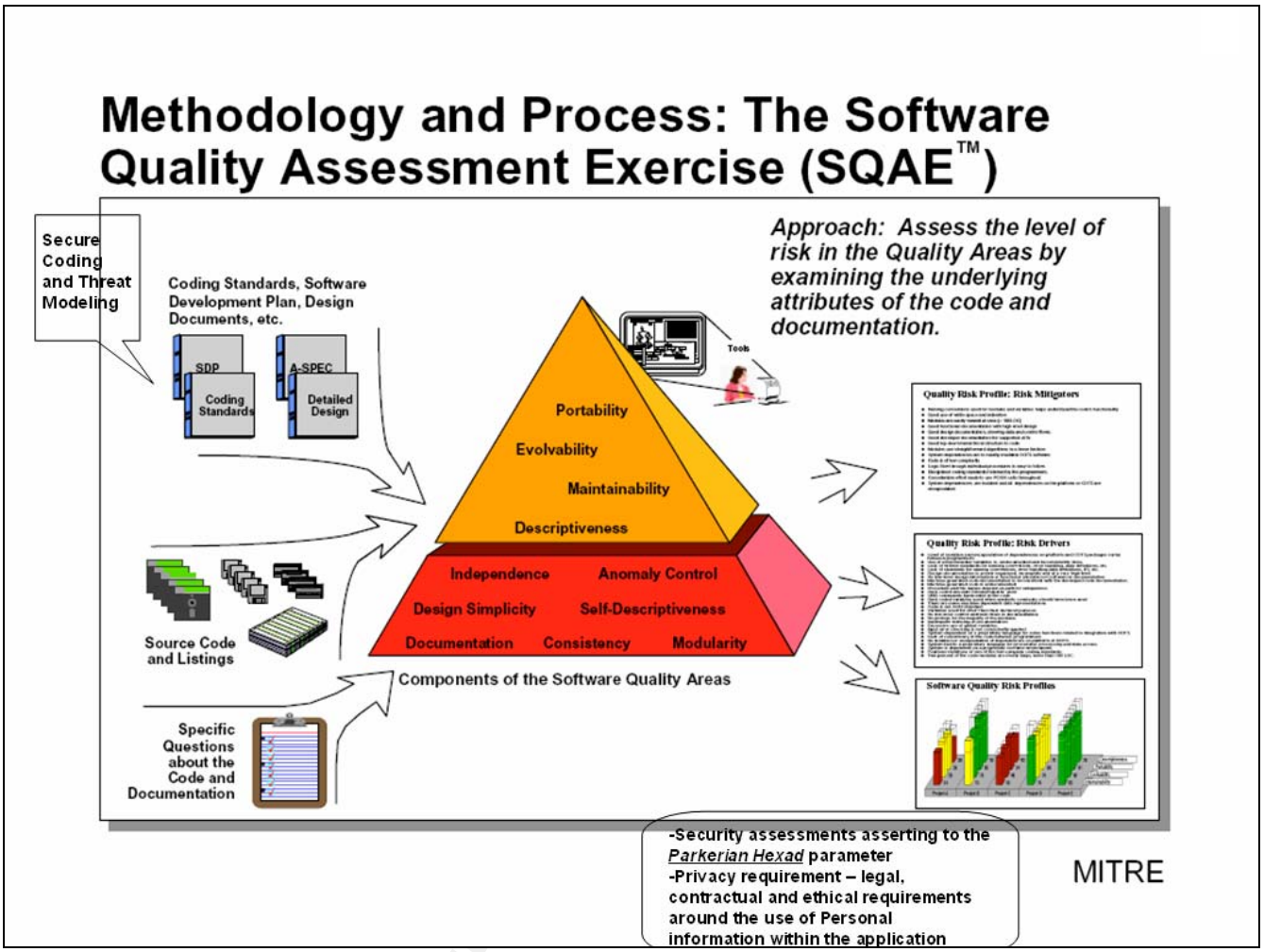

Figure 3 Quality Framework[3] and Security considerations

(Source www.sei.cmu.edu/programs/acquisition-support/conf/2003-presentations/martin.pdf)

Based on the above figure, the security components should be addressing issues that affect the overall software Quality. This framework covers how Security can be plugged into the various measuring factors of Quality like Portability, Evolvability Maintainability and descriptiveness and also considering other factors like Confidentiality, Availability, Integrity, Authenticity, Possession and Utility ("Parkerian Hexad[5]") including Privacy (*Legal aspect in using private information*

within the application should be addressed on how, when and where the Personal information could be used and what ethical requirements are to be followed).

Considering each Security factors is depicted in the table below with an example,

Security Contexts [8]	Description	Example
Confidentiality	Confidentiality refers to limiting information access and disclosure to authorized users -- "the right people" -- and preventing access by or disclosure to unauthorized ones -- "the wrong people."	Identifying the Users and Roles in the application Classify the application data based on the criticality Access Privilege controls in place (Table level, config level etc).
Integrity	Integrity refers to the trustworthiness of information resources. It includes the concept of "data integrity" -- namely, that data have not been changed inappropriately,	Identifying the mode of data transmitted or processed within the application Ensuring components in preserving whatever data processed from corruption/alteration (May

	whether by accident or deliberately malign activity. It also includes "origin" or "source integrity" - i.e. the data actually came from the person or entity you think it did, rather than an imposter	be through Encryption technology) or stricter Session Management during application data processing The concept of Validity or reliability achieved during Application data processing. Periodic monitoring through Integrity checker, Transmission data processing through cryptography systems
Availability	Availability refers, unsurprisingly, to the availability of information resources. An information system that is not available when you need it is almost as bad as none at all	Analysis to identify what information is critical and how it is made available Designing Application related infrastructure and designing the technical requirements

		Replication and Redundancy in Database systems
Possession	The ownership or control of information, as distinct from confidentiality.	If confidential information such as a user-id/password combination is in a sealed container and the container is stolen, the owner justifiably feels that there has been a breach of security even if the container remains closed (this is a breach of possession or control over) Define the roles and responsibilities during analysis phase to determine the ownership of critical components that application data or security controls are dependent on

		Defined access control/privilege list outside the application boundary to the ownership concept
Utility	Usefulness; fitness for a particular use.	If data are encrypted and the decryption key is unavailable, the breach of security is in the lack of utility of the data (they are still confidential, possessed, integral, authentic and available) Analysis phase to determine, how critical components used for security measures are protected (say how cryptographic keys are stored and maintained) and how application is going to implement and use that at what instances etc

		Define the technical aspect of Key Management systems (e.g XKMS, Key Tool etc)
Authenticity	The correct attribution of origin such as the authorship of an e-mail message or the correct description of information such as a data field that is properly named	Repudiation aspect is normally controlled by Cryptographic systems. The Application processing phase (session Management, Authentication scheme, Access privileges list etc) and the SOD concept could leverage the implementation part to achieve this

6. Development and Deployment Phase - Security at its process

- Development Phase activities

Well defined and well-documented coding standards or guidelines form a critical component of "secure" software development process. This would ensure that programmers/developers to follow strictly, certain defined parameters (say rules) while coding rather than programmers developing at their comfort level or preference. Coding standards should be developed on the basis of organizations requirements and standards. This should be achieved or captured during the Analysis phase and once established, the standards can be used as a baseline during coding and evaluation process that could be either manual or automated process with the existing leading practice.

Below is a list of Prioritized factors that could be considered as a security guideline during Development phase.

- **Validation Control both for input and Output data.**
- **Principle of Least Privileges**
- **Define Software Security architecture** - Adhering to Security Policies and Organizational needs
- **Concept of Default Deny** - During defining and implementing access rules and privileges

- **Practice Defense in Depth** - Threat Model and Attack trees well defined and articulated - Documentation with all these details during design phase plays a critical role. Attack trees could be developed based on listing down the different available options of attacks to the application and its related goals. It is required to wear the hat of an attacker while developing attack trees. A sample attack tree[6] could be used a reference for developing based on the existing framework designed

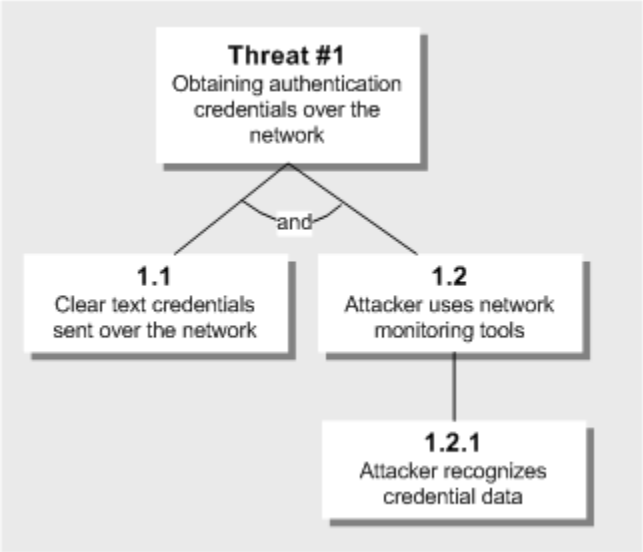

Figure 4 Attack Tree[6] (Sample)

(Reference -)

- Threat Model could be developed based on different areas of security aspect i.e. possible areas of attacks to the application. The Threat model could help to derive probability, the potential harm, the priority of attacks, mitigation factors, impact analysis etc. This could be narrowed down with each application modules defined in the design phase. In one way, incorporating Security into the Engineering

principles can be termed part of Threat Modeling of the application, since it is addressing the various factors of risk mitigation right from Analysis (Requirement analysis phase) till Deployment.

- **Integrating Security into Quality Assurance process** - Refer **Quality Framework and Security Component within the Process** above to derive -

  "Quality Software = Secure Software"

- **Architect and design for security policies.** Create software architecture framework and design software to implement and enforce security policies

- **Design the system as "simple" as possible.** This is to avoid errors since likelihood of errors in complex system during implementation, configuration, and use cannot be ruled out. Additionally, the effort required to achieve an appropriate level of assurance increases dramatically as security mechanisms become more complex.

- **Sanitization of data in Subsystems.** Whenever there is a need for data to be passed between complex subsystems (like command shells, relational databases, and commercial off-the-shelf (COTS) components), ensure sanitizing the data at each end, effectively at the Calling process end since it understands the context it is calling the subsystems modules. Most of the Injection attacks (like SQL, Command) may be able to invoke unused functionality in these components. This need not be necessarily an input validation problem and this could be due to the fact that the complex subsystem does not have a mechanism to understand the context in which the

call is made.

- **Espouse a secure coding standard.** Based on the technology being used, prepare checklist for secure coding.

• Deployment Phase activities

Ensure conducting a Production readiness[7] activity to address the following aspect (risks/issues) of deployment phase activities that is considered to be one of the key phases in getting the developed software in use

- The existing IT infrastructure (Network) does not support the required bandwidth
- Multiple user requests not processed effectively - Server needs to get upgraded
- Insecure configuration of the Application infrastructure possesses high risks to the application being deployed
- Trapdoor (Maintenance hook) created by developers retained in the deployment version of the software
- Ineffective Post - Production processes and strategies leading to business application in risk

All the above mentioned issues create panic and disorder within the development team and management in the entire application development lifecycle. Following points should help to achieve some kind of production readiness during deployment. These steps should start well in advance during the Analysis and Design phase

- Define a team (external from Development team) to have a certification process (internally) to ensure appropriate risks have been addressed during the design and development phase - say design review and code review - to get an unbiased outsider view of the Application
- Create a task force that will help to uncover potential problems that could be show-stoppers. Identifying this at the early stage (pre-deployment) could help to save cost and ensure effective deployment
- Identify the responsibilities for each role identified at the initial stages like Core Team (development, business analyst and testing team), Infrastructure team (back up and recovery, Systems administration and management) and Governance team (business stakeholder, Finance, audit, operations legal, security, standards etc). Cross-team culture and communication play vital role in ensuring technical aspect and business needs
- Testing team ensuring the following tests on the application
    - Load and Performance testing
    - Integration testing
    - System & Functional testing
    - Regression testing
    - Acceptance testing
    - Security testing (hacker's eye on testing application)
    - Post Production factors identification and testing like
        - Disaster recovery/business continuity

- Back up and restore
- Change Management process (e.g. Hardware or Software upgrades)
- Performance monitoring process
- Logging and Auditing (Managed security - processes around those services)

7. Conclusion

Application development should undergo a thorough process engineering framework with security as a process throughout its development lifecycle. Existence of vulnerabilities in a post production application will be a liability for the business owners and will have an unimaginable business impact if the vulnerabilities within the application are exploited.

Security should be considered as a holistic solution rather than considering in isolation. So the question of Security accomplished in QA process or in the pre-post application production process, application penetration testing etc may not be sufficient. Integrated approach of bringing developers, Security professionals, infrastructure team, and business/application owners and finally the end users into the same page is the crucial task in achieving the Organization objective through its business applications

Currently it is becoming clearer on the need of application security, and

standards and regulations started including Application security mandates. Following are some of them,

- IEEE P1074 gives project leaders a plan for including all aspects of the software development life cycle (SDLC) when making security-related decisions[11]
- HIPAA seeks to establish standardized mechanisms for electronic data interchange (EDI), security, and confidentiality of all healthcare-related data[12]
- PCI Data Security Standard deals with Standards included in the requirements of Visa's Cardholder Information Security Program (CISP) and MasterCard's Site Data Protection (SDP)[13]
- Sarbanes Oxley[14] - Systems processing or maintaining financial data need to compliant with the SOX requirement. Most of the financial records are managed (stored/processed/accessed) in electronic format that have application interface components (e.g. Web-based)

These standards and regulations have brought about the necessity of considering application security at its priority to be compliant. Now it's time for Organization's Security Management thought process to reflect on security need into Application development process.

As a conclusion, we should go back defining security from Bruce Schneier's[15] quote that "Security is a Process and not a Product"

8. References

1. OWASP - CLASP (Comprehensive, Lightweight Application Security Process) Best Practice- http://www.owasp.org/index.php/Category:CLASP_Best_Practice
2. Improving Security across Software Development life cycle - http://www.codescan.com/Library/Improving_Security_across_the_Software_Development_Lifecycle.pdf
3. Web Application Security and Sarbanes-Oxley Compliance - http://www.webpronews.com/expertarticles/2006/02/01/web-application-security-and-sarbanesoxley-compliance
4. Definitions from - http://en.wikipedia.org/wiki/
5. Trapdoor - Definition - from http://www.atis.org/tg2k/
6. Risk Management: http://www.computereconomics.com/page.cfm?name=Risk%20Management
7. Software Life Cycle Models - http://www.levela.com/software_life_cycles_swdoc.htm

9. Footnotes

1. Software

Engineering:http://www.levela.com/software_engineering_swdoc.htm

2. Capability Maturity Model:http://www.sei.cmu.edu/
3. Managing Software risks in Software Intensive Systems with Metrics and Measures , Robert A Martin:http://www.sei.cmu.edu/programs/acquisition-support/conf/2003-presentations/martin.pdf
4. Risk Management: http://www.computereconomics.com/page.cfm?name=Risk%20Management
5. Parkerian Hexad:http://en.wikipedia.org/wiki/Parkerian_hexad
6. Create a Threat Model for a Web Application at Design Time: http://msdn2.microsoft.com/en-us/library/ms978527.aspx
7. Are All Systems Go? Production Readiness: http://wsdj.sys-con.com/read/43474.htm
8. Confidentiality, Integrity and Availability (CIA): http://it.med.miami.edu/x904.xml
9. Deployment and Operations: https://buildsecurityin.us-cert.gov/daisy/bsi/articles/best-practices
10. Web Application Security: We need to increase our Budget - http://www.itsecurity.com/security.htm?s=10164
11. IEEE P1074-2005:Roadmap for Optimizing Security in the System and Software Life Cycle - http://www.qualityit.net/Resources/WhitePapers/IEEEP1074-2005-RoadmapForOptimizingSecurityInTheSystemAndSoftwareLifeCycle.pdf
12. How standards and regulations affect application security- http://searchsoftwarequality.techtarget.com/generic/0,295582,sid92_gci1198075,00.html

13. https://www.pcisecuritystandards.org/
14. Sarbanes-Oxley - Financial and Accounting disclosure information - http://www.sarbanes-oxley.com/
15. Bruce Schneier - http://www.schneier.com/crypto-gram-0005.html

10. Glossary

Software engineering is the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of Software

SDLC - Software development Life cycle - Software development process is a structure imposed on the development of a software product. Synonyms include software lifecycle and software process. There are several models for such processes, each describing approaches to a variety of tasks or activities that take place during the process. Capability Maturity Model (CMM) is one of the leading models

Regression Testing - Regression testing is any type of software testing which seeks to uncover regression bugs. Regression bugs occur whenever software functionality that previously worked as desired stops working or no longer works in the same way that was previously planned.

Typically regression bugs occur as an unintended consequence of program changes

Risks - Impact considering (1) the probability that a particular threat-source will exercise (accidentally trigger or intentionally exploit) particular information system vulnerability and (2) the resulting impact if this should occur

Vulnerability - A security exposure in an operating system or other system software or application software component

Trapdoor (Maintenance Hook) - A hidden software or hardware mechanism, usually created for testing and troubleshooting, that may be used to circumvent computer security

Hacker - In computer security, a hacker is a person who specializes in work with the security mechanisms for computer and network systems

COTS - Commercial off-the-shelf (COTS) is a term for software or hardware products that are ready-made and available for sale, lease, or license to the general public. They are often used as alternatives to in-house developments or one-off government-funded developments. The use of COTS is being mandated across many government and business programs, as they may offer significant savings in procurement and maintenance used immediately

11. List of Figures

Last Updated: January 22nd, 2018

Upcoming SANS App Sec Training

Cloud Security Summit & Training 2018	San Diego, CA	Feb 19, 2018 - Feb 26, 2018	Live Event
San Francisco Spring 2018 - DEV522: Defending Web Applications Security Essentials	San Francisco, CA	Mar 12, 2018 - Apr 17, 2018	vLive
SANS San Francisco Spring 2018	San Francisco, CA	Mar 12, 2018 - Mar 17, 2018	Live Event
SANS Northern VA Spring - Tysons 2018	McLean, VA	Mar 17, 2018 - Mar 24, 2018	Live Event
SANS 2018	Orlando, FL	Apr 03, 2018 - Apr 10, 2018	Live Event
Pre-RSA® Conference Training	San Francisco, CA	Apr 11, 2018 - Apr 16, 2018	Live Event
SANS Baltimore Spring 2018	Baltimore, MD	Apr 21, 2018 - Apr 28, 2018	Live Event
Community SANS New York DEV522	New York, NY	Apr 23, 2018 - Apr 28, 2018	Community SANS
SANS Riyadh April 2018	Riyadh, Saudi Arabia	Apr 28, 2018 - May 03, 2018	Live Event
SANS Security West 2018	San Diego, CA	May 11, 2018 - May 18, 2018	Live Event
SANS Northern VA Reston Spring 2018	Reston, VA	May 20, 2018 - May 25, 2018	Live Event
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 OnDemand	Online	Anytime	Self Paced
SANS SelfStudy	Books & MP3s Only	Anytime	Self Paced