

Survey: Security in the System Development Life Cycle

Suhair Hafez Amer¹, Major Jeffrey W. Humphries², Ph.D. and John A. Hamilton, Jr.¹, Ph.D., *Senior Member, IEEE*

Abstract - A general approach to security architecture is introduced. A survey of existing attempts to develop the security architecture introduces the topic. Security can be highlighted as part of the system development life cycle. The authors assume that security cannot be achieved by concentrating on one system component but can be achieved by identifying the relationship between these components and how information is used among them. An original sphere of use and interaction is presented upon which security measures can be evaluated and the required security controls can be chosen.

Index Terms - security architecture, policy, security threats, security attacks.

I. INTRODUCTION

Defining a general structure for a security architecture is challenging because each organization attempts to tailor security according to its needs. To develop a secure system, a thorough understanding of the desired security properties and functional and non-functional properties of the system is required.

Typical attempts to define a security architecture are concentrated on one system component and ignored by the rest. Such an approach was tailored to the organization's needs. For example, [1] proposed a three-step approach to secure architectures. First, the system architecture is formalized in terms of common architectural abstractions. Then the system architecture is refined into specialized architectures where each one is suitable for implementation under different security assumptions. Finally, a proof is constructed to see if every implementation satisfies the intended security policy.

Another approach was proposed by [2] who recommend employing six layers of security to protect the operations of an organization. The *physical* security layer addresses the protection of physical items, objects or areas from

unauthorized access and misuse. The *personal* security layer addresses the protection of the individual or a group of individuals who are authorized to access the organization and its operation. The *operations* security layer focuses on the protection of detail of a particular operation or series of activities. The *communications* security layer encompasses the protection of the organization's communication media, content and technology. The *network* security layer protects the network components, connections and contents. Finally, the *information* security layer is concerned with protecting the information, the systems and hardware that use, store and transmit that information.

In this paper, the authors will attempt to highlight security architecture development through the steps that are carried out during the system development life cycle. The steps and the details of each step to be considered while developing secure architectures can be found in [2].

II. INVESTIGATION PHASE

Investigation is the first phase in the development of the security system life cycle. In general the project scope and goals are outlined in this phase and the existing resources are evaluated. Security principles and practices can help in developing and identifying the goals that are to be accomplished while trying to secure the system. The result of this phase is documenting the goals in the program security policy [2].

A. Evaluate Existing Resources

Identifying the system requirements is important because it determines the specification of the security policy. In general, any system consists of five components [2] that enable the inputting, processing, outputting and storing of information. The first component is *software* that corresponds to the applications, operating systems, and the assorted command utilities. The software components are the most difficult ones to secure since they are developed under demanding constraints and are not

1. Auburn University, Auburn, Alabama.

2. United States Air Force Academy, Colorado Springs, Colo.

checked fully for security leaks, holes, bugs, or weaknesses. The second component is *hardware* which is the physical technology that hosts and executes the software. The hardware component also stores data and provides the interfaces for information entry and removal. The third component is *data* which is stored, processed and transmitted through the computer system. The fourth component is *people* who can pose a threat on information security intentionally or unintentionally. Finally, *procedures* are the written policy instructions that accomplish a specific task.

B. Security Principles and Practices

Stoneburner, Hayden and Feringa present a compiled list of engineering principles for system security by the National Institute of Standards and Technology [3]. These principles do not apply to all systems at all times and therefore should be carefully considered throughout the lifecycle of a system. This is also not an inclusive list due to the constant changes in the information system security environment.

C. Security Goals

Organizations have different security goals but generally those goals include confidentiality, integrity and availability [4]. The first goal is confidentiality which ensures that any computer assets are to be accessed only by authorized parties. Integrity ensures that modification of assets can only be performed by authorized parties and in authorized ways. Availability ensures that authorized parties may always have access to their assets at authorized times which is opposite to denial of service. Availability applies, in general, to data, systems and services that are considered available if there is a timely response to requests and a fair allocation of resources. Also availability implies that the system is fault tolerant and will gracefully shutdown in case of a hardware or software fault. Another point of view of availability states that a system or service can be easily used in the way it is intended to.

Whitman and Mattord define additional security goals [2]. Information accuracy is information free from mistakes or errors and has the value that the end user expects. Authenticity is the state of being original and not a reproduction or fabrication. Utility of information is concerned with the quality or state of the information having value for some purpose or end. Information should serve a particular purpose and should be available in a meaningful format to the end user. Finally, possession of information is the quality of having control or ownership over the information.

D. Security Policy

It is important to define the security policy before developing the security architecture. Ultimately, the validity of the security architecture is measured against the security policy. The security policy is to identify the security objective, the participating entities and the underlying assumptions. The policy consists of rules followed by the user and also the identification of security objects or resources and the relationship between them and the users [5]

Security policies are used for several purposes such as recognizing sensitive information assets, clarifying security responsibilities of users, promoting the awareness of existing employees and finally guiding new employees to the goals and constraints of using a system [6].

Pfleeger and Pfleeger identify four characteristics that make a good security policy [6]. A security policy must be comprehensive and apply to all possible situations and should allow natural expansion to new cases. Secondly, a security policy should be durable in the sense that it should grow and adapt well and have no ties to any specific data or protection mechanism. Third, a secure policy must be realistic in the sense that it must be possible to implement the required security with the available technology. Finally, it should be useful in the sense that it should not be obscure or incomplete because it will not be implemented properly.

A policy should be tailored to fit the system that is to be secured. For example, a single security policy is not practical for a distributed system and it is preferred to have each distributed component enforcing its own security policy [7].

The National Institute of Standards and Technology proposes the use of three types of security policy [2]. The first is the *general or security program policies*. This is a document that guides the development, implementation and management of the security program, it has the requirements that must be met by the information security framework and defines the scope, purpose, constraints and applicability of the security program in the organization. It also assigns responsibilities for various areas of security and finally addresses legal compliance. The second is the *issue-specific security policy*. This is used to instruct employees to use the various technologies and processes that are used to support routine operations properly. Issue-specific security policy addresses specified areas of technology, requires frequent updates and contains a statement of the position of the organization of a specific issue. The third is the *systems-specific security policies*. These are codified as standards and procedures that are used when configuring or maintaining systems.

An organization has the choice to apply one policy but it is preferred to use a hyper policy that better fits its security needs [6]. The Clark-Wilson security policy also called well-formed transactions specifies that steps should be performed in order and the steps performed should be exactly the ones that are listed and the individuals that perform the steps should be authenticated. In general, the purpose of this model is to maintain the consistency between the internal data and the external user expectations of those data. The *separation of duty commercial security policy* is concerned with separation of responsibility or duties. This is accomplished manually by using dual signatures. Finally, the *Chinese Wall commercial security policy* is concerned with the need for creating commercials for information access protection. It is mainly relevant to people who might be subject to conflicts of interest such as those in legal, medical, investment or accounting firms.

III. ANALYSIS PHASE

In the analysis phase the current system is assessed against the plan developed in phase 1 and the organization develops the preliminary system requirements. The plan should describe how to integrate the new system with the existing system and documents its findings and update its feasibility analysis. At the same time the plan should contain analysis of its existing security policies and programs and the existing threats and controls. Finally, the plan should examine the legal issues and perform risk analysis to its resources [2].

In order to secure a system, threats or attacks should be identified and classified. Stallings identified a threat as being a set of circumstances that has the potential to cause harm or loss to a computing system [4]. They grouped threats into four general categories that are *interception, interruption, modification* and *fabrication*. However, [2] identified five threat groups. The first group is an *inadvertent act* in which the malicious intent is absent or can not be proven. The second group is *deliberate acts* which includes and is not limited to the acts of espionage or trespass, shoulder surfing, information extortion acts of theft of physical, electronic, or intellectual properties of an organization or software piracy. The third group consists of *Acts of God*. The fourth is a result of a *technical failure*. And finally, the fifth is a result of *management failure* especially when management lacks sufficient planning and foresight to anticipate what technology is needed to the evolving requirements of the organization.

A. Attacks exploit component vulnerabilities

Network devices are vulnerable because it is easy to identify and see the devices that are connected to the system. Adding, changing, removing, intercepting traffic

to and denying service to the devices connected to the system are attack strategies. Furthermore, hardware may suffer accidental acts that are not intentional “involuntary machine slaughter” or “voluntary machine slaughter” which is when a person actually wants to harm the hardware of a system [6].

Software can be defined as the operating system, controllers, utility programs, or application programs that can be used on computing equipments. It can be replaced, changed, maliciously destroyed, modified, deleted, or misplaced. Examples of software vulnerabilities are: *software deletion, software modifications, and software theft* are presented in [6] and [2].

Both hardware and software attacks are not easily interpreted by the general public as data attacks. This is because data has a visible nature and can be understood when put in context. In many cases, the sensitivity of data declines over time. Examples of data vulnerabilities [2] include but are not limited to forcing access to a data store. Common forced entry techniques include *password crack, brute force, and the dictionary attack*.

A more detailed list of network vulnerabilities can be found in [8] and [2] and which include the following vulnerabilities: *spoofing, spam, mail bombing, sniffers, social engineering, masquerade, cookies, scripts, traffic analysis, passive eavesdropping, active eavesdropping, unauthorized access, man-in-the-middle attack, session high-jacking, replay, active content, buffer overflows, dot-dot and address problem, application code errors, server-side include, denial of service, distributed denial of service and e-mail threats*.

In databases, data as well as their characteristics are considered to be sensitive. One example is data disclosure where the user may know the exact data or value of data by accident after requesting it without knowing that it contains sensitive data. In this case it is the responsibility of the database manager to make sure that the data will not be displayed to the user. Some users may be able to know the value of the field by knowing first the bounds of it. Using a simple narrowing technique the user is able to find as a result the actual value of the field. Negative result is the technique used by some users to know the value of a field by using a simple query that will determine a negative result. In some cases knowing the existence of a field is itself a sensitive piece of information and which might affect user behavior. Finally, some users may be able to find the probability of a value by asking a list of questions related to the field of interest and then interpreting the answers to find the value.

IV. LOGICAL DESIGN PHASE

In the logical design phase the organization needs to assess current business needs against the plan developed in phase 2. The organization is now required to select the applications to be used and the data and structure support. At this stage the security blueprint is developed and plans for the incident response actions and business response to disasters are highlighted. Finally, the feasibility of continuity and / or outsourcing of the project should be determined [2].

A. Security Models

Models are used to describe, study or analyze a particular situation. They are used in general to test if a particular policy is complete and consistent. They are also used as a way to document a policy. They can aid in conceptualizing and designing an implementation. Finally, a model is used to check if the implementation meets the requirements [6]. Next are some examples of security models.

The *security model* of the National Security Telecommunications and Information Systems Security Committee (NSTISSC) is a comprehensive model for information security and is becoming the evaluation standard for the security of information systems [9], [10].

The *lattice model* of access security is a generalized model on which the security model is based. It is based on the concept of a lattice which is a mathematical structure of elements organized by a relation among them. In general, its representation of increasing degrees and sensitivity levels can apply to many computing situations [6].

The *Bell-La Padula Confidentiality Model* is mainly used to identify the allowable communication while maintaining security and is a formal description of the allowable paths of information flow in a secure system. This model has been in general used to define security requirements for system that are handling data concurrently at different sensitivity levels and is mainly used to identify the paths that could lead to inappropriate disclosure of information [11].

The *Biba Integrity Model* was constructed mainly to prevent inappropriate modification of data. Biba Integrity model defines the integrity levels that correspond to the sensitivity levels defined in Bell-La Padula model [6].

B. Security blueprint

To achieve maximum security several layers of security should be employed. *Physical Security* is the process of knowing what aspects of the computing environment will

or have an impact on security. It is in general used to describe the security needed outside the computer system [6].

Network Security is concerned with designing a secure network perimeter and policies that require addressing issues such as: knowing whom are the potential adversaries, understanding the work environment and how things usually operate [12]. The cost of applying security to the systems should be evaluated. The organization should know their security weaknesses, how they can be exploited and then study carefully any assumptions regarding security. Finally, the organization should be able to control its secrets and limit the scope of access and trust.

Furthermore, to create network security it is important to create a strong network perimeter that protects internal resources from outside threats. Three types of perimeter networks can be used: the outermost perimeter, internal perimeters and the innermost perimeter. A successful network security perimeter must have a firewall and a gateway for all of its communications between the trusted networks and the un-trusted, unknown and semi-trusted networks [12].

Information Security can be achieved by using a hybrid framework [2]. Their sphere of use shows the different ways users can access information and that user and system have direct access to the information, however, networks have an indirect access since a person accessing the internet must go through the local network and then access the system that contains the information. Their sphere of protection shows that there must be a layer of protection between each layer to prevent access to the inner layer from the outer one which reinforces the concept of defense in depth.

C. Modified security blueprint

Figure 1 represents a use and interaction model of system components. People may have access to data and information directly or through the use of DB, networks, systems or internet. The use can be governed by specified procedures and policies or can be used with out them. Systems are divided into software and hardware components, each having a list of vulnerabilities that can be exploited. Data do not need to be kept electronically to pose a threat. Much information is kept on hard copies and some are in the minds of authorized personnel. Measures should be taken to protect such information since security breaches in most cases happen from human disclosure to such information.

Previous attempts to secure architectures concentrated on one component or aspect of the interaction and use model. To achieve security, an organization should make sure

that all components are secured properly. For example, one organization may concentrate on securing the interaction between people and networks since it poses a constant threat but if DB security is ignored a more serious and more dangerous security breach will be present.

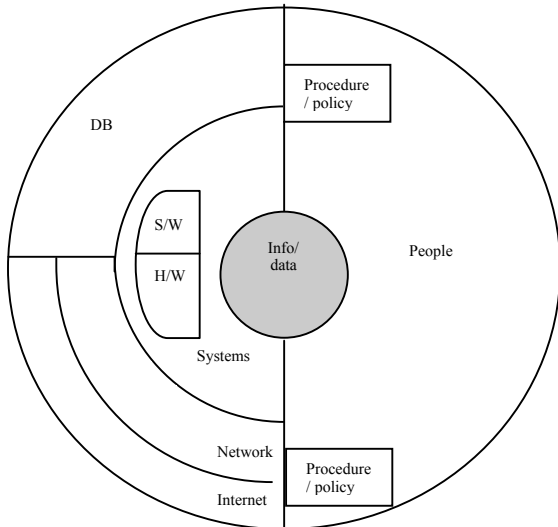


Figure 1: Sphere of use and interaction between system components.

V. PHYSICAL DESIGN PHASE

In the *Physical Design* phase the organization needs to select technologies to support solutions developed in phase 3. At this phase the organization chooses the best solution and decides accordingly to either buy or make the required components. The findings should be documented as usual and the feasibility analysis updated. Supports should be provided to the security blueprint and a successful solution defined. Physical security measures are designed to support technological solutions and finally the project is reviewed and approved [2].

A. System security Mechanisms

Different security technologies are generally available in the marketplace. Security architecture components include both hardware and software elements.

B. Network Security Mechanisms

Another important technology is the wireless security technology which can be divided into three broad categories: authorization, maintaining the privacy of the session and verifying the integrity of the information. Welch and Lathrop provide a compiled list of some mechanisms used to secure network transmission [8]. Additional approaches to add security to a system can be found in [13].

C. Database Security Mechanisms

Database management system is responsible for handling the integrity, confidentiality and availability of data on three dimensions. Database integrity is concerned with the protection of the entire database against damage or the corruption of the master database index. This can be addressed by operating system integrity control and recovery procedures. Element integrity is achieved by using the proper access control to protect a specific data element from being changed or written by unauthorized users. In order to ensure element accuracy, checks on the values of elements can be used to prevent the insertion of improper values whereas constraint conditions can be used to detect incorrect values. Many databases maintain additional information in order to detect inconsistencies. One way for a database to recover data is by maintaining a log of users' accesses and what they have changed. Therefore, in the case of a failure the database backup is reloaded and all changes are applied from the log file. The concurrency/consistency problem resulting from many users accessing or sharing the same database can be solved by using different kinds of locks. Finally, a monitor can be used to check the integrity of the data being entered and that it is consistent with the rest of the database characteristics [6].

Multilevel databases require a different type of security measures. In general, multilevel databases differ from regular database in that first the security of one element may be different from the other elements in the same row or columns. As a result security must be implemented for each individual element. Second, representing security situations only using two levels of security that are sensitive and non-sensitive is inadequate. This is because in many cases several grades of security may be needed. Finally, the security on an aggregate value may be different from the security of individual elements. In order to ensure that each element is associated with a related sensitivity level a control policy should be accessed indicating the privileges of each user and there should be a mean to ensure that this value was not changed by unauthorized user. One way to limit access is the use of separation which can be implemented using partitioning, encryption, and the use of integrity and sensitivity locks. Some examples of multilevel secure databases are: integrity lock model, trusted front-end model, commutative filters model, distributed databases and window/view based model as noted in [6].

D. Software Security Mechanisms

Usually memory protection and privilege levels have been implemented in hardware. However, recent mobile code systems rely on software rather than hardware for

protection. Such software mechanism choices are being driven by two needs: portability and performance [14].

VI. IMPLEMENTATION PHASE

In the implementation phase the organization develops or buys the required software as well as ordering the required components. It is preferable to document the system and the system is presented to the users and then they are allowed to start using it. The feasibility analysis is updated as usual. At this stage the system is being tested and its performance is being reviewed. Finally, the organization can buy or develop security solutions at this phase [2]. In general at the end of the implementation phase, a tested package is presented to management for approval and the secure system is developed and validation, verification and evaluation should be performed on the system.

A. Validation, Verification and Evaluation

Most system consumers are not security experts and in need of security functions but are incapable of verifying and validating that the system correctly implants the security policy. This is why it is essential sometimes for the consumers to have an independent third party to evaluate the security of the operation system.

Three evaluation procedures are highlighted in [6]. The U.S. "Orange Book" Evaluation / U.S Combined Federal Criteria was published by the U.S. Department of Defense (DoD) which defines a set of distinct, hierarchical levels of trust in operation systems. The information Technology Security Evaluation Criteria known as European ITSEC Evaluation was a European effort to develop a criterion and methodology for evaluating security-enforcing products. Finally, the Common Criteria is a result of the efforts of both the United States and Canada. It has the concept of security targets and protection profiles.

VIII. MAINTENANCE AND CHANGE PHASE

Finally, in the maintenance and change phase, the system is supported and modified as needed. The system should be periodically tested for compliance with business needs and upgraded and patched as necessary. Finally, the system should be constantly monitored, tested, modified, updated and repaired to meet changing threats [2].

After implementing and installing the secure system, security should be maintained by monitoring the security triple: threats, assets and vulnerabilities. Since many factors can shift the information security environment, a security management model should be adopted in order to assist in managing and operating the ongoing security. Whitman and Mattord introduced the modified ISO

Network Management Model that is a five-layer approach that provides structure to administration and management of networks and systems [2].

VIII. CONCLUSION

The modified sphere of use and component interaction model suggests that in order to achieve security, all the components of model should be considered while addressing security during the developments life cycle. The interaction between every component: people, DB, System, information, network, internet, and procedures and policies can pose a threat to security. One should not focus on just protecting network and internet components, for example, because the raw data or data in large scale databases must also be protected.

In this paper the authors presented a new sphere of use diagram explaining that threats can exist at different levels and components. Security has been introduced and surveyed while being employed throughout the development life cycle that is used by any organization and how security can be employed had been explained.

IX. REFERENCES

- [1] Moriconi, M., Xiaolei, Q., Riemenschneider, R.A., and Li, G., "Secure software architectures", Proc. IEEE Symposium on Security and Privacy, 1997, pp. 84 - 93.
- [2] Whitman, M. E., and Mattord, H. J. *Principles of Information Security*, Thomson Course Technology. Canada 2003.
- [3] Stoneburner, G., Hayden, C., and Feringa, A. "Engineering Principles for Information Technology Security (A Baseline for Achieving Security)" NIST Special Publication 800-27, June 2001.
- [4] Stallings, W., *Network Security Essentials*, Prentice-Hall, Upper Saddle River, New Jersey, 2000, pp 6 – 11.
- [5] Foster, I., Kesselman, C., Tsudik G. and Tuecke S. "A Security Architecture for Computational Grid," Proceedings of the 5th ACM Conference on Computer and Communications Security Conference, San Francisco, California, November 1998.
- [6] Pfleeger, C. P., Pfleeger, S. L., *Security In Computing. Professional Technical Reference*, Prentice Hall, Upper Saddle River, NJ, 2003.
- [7] Gasser, M., Goldstein, A., Kaufman, C., and Lampson, B. "The Digital Distributed System Security Architecture," Proceedings of the 1989 National

Computer Security Conference, Baltimore, October 1989, pp. 305-319.

[8] Welch, C., Lathrop, S. "A Survey of 802.11a Wireless Security Threats and Security Mechanisms," A Technical Report to the Army G6, 2003.

[9] "NSTISSI No. 4011 – National Training Standard for Information Systems Security (INFOSEC) Professionals." 06/1994. www document. Viewed 02/12/2002.
<http://www.nstissc.gov/Assets/pdf/4011.pdf>.

[10] "NSTISSI No. 4014 - National Training Standard for Information Systems Security Officers (ISSO)." 08/1997. www document. Viewed 02/12/2005.
<http://www.nstissc.gov/Assets/pdf/4014.pdf>.

[11] Bishop, M., *Introduction to Computer Science*, Addison-Wesley, Boston, 2005.

[12] Campbell, P., Calvert, B., and Boswell, S. *Security + Guide to Network Security Fundamentals*, Thomson Course Technology. Canada 2003.

[13] Spencer, R., Smalley, S., Loscocco, P., Hibler, M., Andersen, D. and Lepreau J. "The Flask Security Architecture: System Support for Diverse Security Policies," Proceedings of the 8th USENIX Security Symposium, pages 123--139, Washington, DC, August 1999.

[14] Wallach, D. S., Balfanz, D, Dean, D. and Felten, E. W. "Extensible Security Architectures for Java," Proceedings of the Sixteenth ACM Symposium on Operating System Principles, pages 116--128, Saint Malo, France, October 1997.