Fall 2007

Security architecture methodology for large net-centric systems

Njideka Adaku Umeh

Department:

SECURITY ARCHITECTURE METHODOLOGY

FOR LARGE NET-CENTRIC SYSTEMS

by

NJIDEKA ADAKU UMEH

A THESIS

Presented to the Faculty of the Graduate School of the

UNIVERSITY OF MISSOURI-ROLLA

In Partial Fulfillment of the Requirements for the Degree

MASTER OF SCIENCE IN SYSTEMS ENGINEERING

2007

Approved by

Ann Miller, Co-Advisor	**Cihan Dagli, Co-Advisor**

Jagannathan Sarangapani

ABSTRACT

Conceptual architectures provide a means for architecting complex network centric systems. This approach provides the basis for security architecture for a network enabled capability. Security assessments and security architectures are of utmost importance in any system development to ensure that systems are properly protected from both unauthorized access and or malicious applications, especially for network enabled systems as societies' use of such system approaches an all time high. Therefore a need for a structured approach in designing such systems is necessary and has been recognized. Of particular importance is a methodology that cuts across a wide variety of security tasks and needs. By assessing the challenges, relevance and requirements a security architecture methodology is put forward to take care of the security needs of a complex network centric system. This thesis describes an over-arching security architecture methodology for large network enabled systems that can be scaled down for smaller network centric operations such as present at the University of Missouri-Rolla. By leveraging the five elements of security policy & standards, security risk management, security auditing, security federation and security management, of the proposed security architecture and addressing the specific needs of UMR, the methodology was used to determine places of improvement for UMR. Conclusions and future works were also highlighted.

ACKNOWLEDGMENTS

I would like to thank all who made this study possible. First, I would like to thank my advisor, Dr. Ann Miller, for her continuous support and supervision during my master's study at UMR. I would also like to thank Dr. Cihan Dagli and Dr. Jagannathan Sarangapani for their guidance and support. I am also grateful to the following:

The folks at Institutional Research & Assessment office Emily Petersen, Melissa Folsom, Billy Earney and the former director Dr. David Saphian. Stuart and Martina Baur their kids Markus, Erich and Wilhelm; grandparents Omi & Opa and Oma & Opapa, thanks for letting me share in the joys of your family.

My parents Ugonma & Ngozi Umeh and my siblings, Chidinma Ihebom & family, Jideofor Umeh, Ogadinma Umeh and other members of my extended family, especially Grace & Anthony Okafor for their love and help.

My friends Emel Meteoglu, Nil Kilicay Ergin, Oyku Selimoglu, Lea Dankers and Firuze Duygu Caliskan, Michael Taft and my colleagues at the Smart Engineering Systems Lab. The Engineering Management and Systems Engineering Department for providing funds during my study at UMR.

And most of all I thank God for his mercies.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF TABLES

1. INTRODUCTION

In describing a security architecture methodology, several relevant concepts emerge. These concepts include trust, policy and standards, services, risk management and so on. Also, these concepts in themselves have enormous application and span a broad spectrum, across various disciplines.

However, in this thesis, these concepts are defined or represented in the context of security in information technology networks, which is the bed rock of net-centricity.

Furthermore, net-centricity or network-enabled capability (NEC) which society have come to depend on is built on strong architecture; unfortunately it is designed with the intention of being open. This places the responsibility of securing the networks and information to the individual users or organizations, ergo the need to develop a security architecture framework or methodology that can easily be adapted and scaled for various systems.

The recent progress achieved in information technology has increased the connectivity within societies, governments, industries and individuals. This has largely been made possible through internet and broadband communication technology advances; which in turn have created systems that are very different from past systems. Network-centric is the term used to describe such systems. These systems comprise a diverse category of large and complex systems whose purpose is providing network-type services. Network-centric systems are also frequently collaborative systems that are built on partially voluntary and uncontrolled interaction of complex elements in an ad hoc environment [1].

A security architecture methodology is needed as systems become more complex and multi-disciplinary. Network-centric systems are complex not just because of the number of subsystems and components present, but also because of real time data components required and the diversity involved. Some examples of network centric systems are the military systems, banking system and the educational systems. All these systems and their respective industries have come to rely heavily on information technology and the net-centricity it affords. This further emphasizes the critical need for

security architecture for these systems and invariably a security architecture methodology.

Therefore, there are challenges in prescribing an architectural methodology for the security of such a system, which will accommodate the subsystems complexity and real-time data distribution capabilities as well as the diversity associated with them. In order to avoid system degradation resulting from failure to handle large amounts of real-time data distribution, complexities in net-centric systems need to be addressed.

1.1. MOTIVATION

Going through several papers searching for information in security architecture, it became apparent that there was no paper addressing security architecture methodology, especially a step by step application for a network-enabled capability. Therefore, this thesis aims to put forward a methodology that can be applied for both large and small net-centric systems.

A security architecture methodology should include physical protection, network security, message-level security and application-level security. In addition, should include security in the form of authorization, privacy, policy, trust, and secure conversation. Lastly the methodology should accommodate a federation of systems.

1.1.1. Goals of Security Architecture Methodology. At the beginning of this thesis, the following goals for a security architecture development methodology were established in order to set a clear focus for the paper.

(1) The methodology should efficiently and effectively facilitate the development of an integrated security architectures and plans for enterprise networks.

(2) The methodology should facilitate an integrated solution across very complex heterogeneous information systems.

(3) The methodology must result in a final product that covers all customers' requirements, policies and guidelines.

(4) The methodology must provide the customer with visibility into the process as well as the solutions.

(5) The methodology needs to facilitate incorporation of changes due to advances in technology and revisions of policies and guidelines.

1.1.2. Relevance of Security Architecture to NEC. In order to have a perspective of what the task process will entail, an attempt is made to discuss some of the reasons security architecture is necessary for a network-enabled capability (NEC) under the following high- level groupings.

Information Sharing

Information sharing is fundamental for success in both every day lives and business ventures; however it is important that the information that is sent be secured, that is, the information be retrieved and viewed by only the intended parties.

Security architecture therefore supports information sharing by assuring users that their communications will be private. Security architecture establishes a common language for security. It provides structure which can facilitate learning and understanding of the complexities of the security requirements and solutions and their interactions with the rest of the system. It provides the industry a coherent way of how elements of the security solution must fit into the overall.

Interoperability

Net-centricity systems suppose a federation, a no single owner situation and interoperability of all these systems is required to make it work. Security architecture provides a coherent technical approach to security across any organization that would enable secure interoperability. It would enable the representation at any time the best knowledge on technical security solutions.

It would help present security problems and solution as an easily assessable and identifiable process which could be described at different levels of detail i.e. from high level to lower levels of implementation.

It would show some progress in prototyping scenario which can be validated by implementing any part of a security architecture using the methodology provided

Uninterrupted Operations

Services and businesses that have come to depend on net-centricity work most exceptionally because of their uninterrupted operations. This has come to mean that the infrastructures that run on a net-centric environment are critical and needs a security measure to protect it. Security architecture therefore provides a guideline for the protection of all these critical infrastructures.

Security Risk Management

Security architecture would provide a coordinated security risk management that would provide guidelines for decision making and, as a result, would optimize the investment in security resources.

A security risk management process would provide a tool which will help in the implementation of the chosen risk analysis methodology. The input will be continuously changing threat, vulnerability and impact (or asset value) information and the output will show the requirements for security countermeasures of all types (e.g. physical, personnel, procedural and technical) to maintain risk at acceptable levels.

It would provide for an informed risk management decisions to be made by allowing accreditors to identify all the security implications caused by a change to any element of the infrastructure.

It would help in identifying security risks for a NEC for which countermeasures might need optimizing or are yet to exist.

From the above potential beneficiaries can be derived such as stakeholders, security risk owners, capability managers, accreditors, system managers, security (including cyberdefence) managers, researchers and developers, system designers, security product designers, product vendors and even end-users who will use and benefit from the security architecture.

1.2. PROBLEM DEFINITION

Some challenges that would be encountered in the design of security architecture and its methodology for a large net-centric system are put forward in this section.

1.2.1. Challenges of Prescribing Security Architecture for NEC. In prescribing security architecture for a NEC, challenges are faced. Some of the challenges are unique to information technology/NEC while the others are general. Some possible requirement for security architecture have been listed above, possible challenges that might be encountered are mentioned subsequently.

Firewall Limitations: – It is of import to know the limitations of firewall which may not detect potential attacks. Examples are attacks that are disguised to cause internal application buffer overflow. New firewall products designed to protect Web Services at

the Simple Object Access Protocol (SOAP) level are in the market now, however it has yet to be determined how effective they are and also their position within the security architecture is not yet clear.

Service-Level Security Semantics: – Definitions and standardizations are yet to permeate the fabric of security. The standards for the mechanism by which different parties interface with each other to achieve security goals such as authentication and authorization are yet to be defined and prescribed.

Interoperability of Security Solutions: –The lack of standard at the service interface level, has made it impossible for security products currently in the market currently to be fully interoperable.

Security vs. Performance: – The use of public key (PK) encryption in security architecture requires computation intensive tasks that include message signing, encryption, and certificate validation. In this instance sending a secure message will be several times much slower than a less secure version, and a direct inverse relationship between performance and security can be computed. Therefore cautious planning and effective optimization techniques are necessary to meet operational requirements.

Impacts on Existing Policies and Processes: – In addition to the identification of system boundaries, trust relationships need to be established for a more dynamic application in a net-centric environment. This will require the establishment of trust domain relationships.

1.2.2. Other Challenges. In addition to the above, the challenges below need to be considered, especially keeping in mind that security needs to be dynamic since it has become increasingly clear that the term “security” means nothing unless[2] it is possible to know who needs to keep out and for how long they need to be kept out.

The idea of a security architecture methodology for net-centric system not only implies the governance and maintenance of the architecture but that dynamic security policies and policy based access control be incorporated into the architecture and methodology e upgrades. This will furthermore entail more challenge in achieving an architecture which is sufficiently adaptable in its implementation of the risk management methodology and accommodate future uncertainties. If the security risk management processes are modeled incorrectly in the higher levels of the architecture, the problems

will be cascaded to lower levels. In this thesis an attack tree model is used to demonstrate the importance of risk management.

It is also important to note that a user-friendly architecture will be of concern since the architecture will be used by a wide variety of people most of whom may not be versed in all the technical details.

A security architecture for net-centric system will need to be achieved using a Service Oriented Architecture (SOA) with “ubiquitous security services” being applied at all OSI layers [3]. Melrose and Madahar [4] uses an OSI/ISO layer type architecture to describe an overarching security architecture for NATO NEC which can be incorporated in this methodology description.

1.3. SECTION ORGANIZATION

In here describes the structure for the rest of this thesis. Section 2 is the literature review. It presents a brief summary of security architectures of the Department of Defense Information System and the NATO, the standards and heuristics tool that help in systems architecting, and document of the Joint Technical Architecture- a DoD document that specifies technical standards for interoperability. It also discusses research in complexity theory, architecture overview- by presenting the different types of architecture, security architecture and security architecture, architecture frameworks and network enabled capability.

Section 3 discusses the methodology of a security architecture -by looking at the requirements, critical success factors for a security architecture methodology, and delves in details on the actual methodology.

Section 4 presents an example of security architecture for net-centric system using the University of Missouri-Rolla by adopting the methodology already prescribed in section. Finally, Section 5 presents conclusions and future work.

2. LITERATURE REVIEW

In recent times, all systems are distributed to some degree. They include[5] communication system allocating channels through distributed switches, aircraft flown using distributed controls, computer memories built out of hundreds of thousands of distributed active elements, banking systems and credit card services operating at different locations serving customer several million miles away, military operations relying on distributed elements, all operating in whole or part through network enabled capability. Unfortunately, as societies have come to rely heavily on information technology and the advances it affords by way of net-centricity, it becomes increasingly important that net centric systems such as these systems mentioned should have security architecture in place to help in the countering of unauthorized access to systems.

Hence, the criticality and the urgency of security architecture, ergo a methodology for such distributed/net-centric system cannot be over emphasized. It goes without saying that because of societies reliance on these system, an outage due to an attack will be costly, both in monetary terms and otherwise.

Even though societies have achieved stupendous technological advances, infrastructures are still prone to security flaws ever more so now, than in previous technologies. The difference between previous security measures and current security architecture for NEC is that, it has become much easier to crack security because of technology. For instance a safe that recognizes and stores the index finger to unlock it is much easier to crack because if the finger impression is not wiped, using any other mould to make the impression will unlock the safe.

Applications for this thesis extends from civilian operations such as in the energy sector (pipeline monitoring of petroleum products), telecommunications sector (secure communication for customers), to military operations.

There was little information in literatures regarding security architecture methodology for large scale net-centric systems; however various literatures on security, security architecture, net-centric system, complexity and more were reviewed; from which information has been gathered on what security architecture should consist of. A step by step method of prescribing this security architecture is therefore needed to

provide both architecture designers and implementors alike a guide of how and what is necessary to build security architecture successfully in systems that are network enabled.

It is important to note that network enabled capability and network centric/net-centric are used interchangeably in this thesis to refer to activities across and within a network of participants in a continuously-evolving, complex community of people, devices, information and services interconnected by a communications network to achieve optimal benefit of resources and better synchronization of events and their consequences. This entails allowing a wide range of people including entities that are known and are trusted as well as strangers, access to networks which might be misused in a manner that threatens the network. Furthermore, the challenges mentioned previously are compounded by the fact that complexities existing in net-centric systems need to be address due to real time data demand and distribution especially in military operations, therefore the discussion and understanding of complexity theory.

Complexity can be described as the degree of difficulty to understand, verify and formulate system behavior; even when knowing information about the components, their numbers, arrangements, functional properties and interactions. Edmonds [6] writes that complexity is that property of a language expression which makes it difficult to formulate its overall behavior even when given almost complete information about its atomic components and their inter-relations. Complexity can also be defined as a measure of uncertainty in achieving a set of specific functions or functional requirements [7]. This leads to the concept of securing a complex network of people using the information technology infrastructure.

2.1. ARCHITECTURE OVERVIEW

The DoD Goal Security Architecture (DGSA) [8] [9] defines four types of architecture. DGSA was developed as part of the technical Architecture Framework for Information Management (TAFIM). The four types of architectures are discussed overleaf.

Abstract Architecture

The abstract architecture defines principles and fundamental concepts that guide the selection and organization of functions. This level of architecture cites principles, fundamental concepts and functions that satisfy the typical security requirements.

Generic Architecture

The generic architecture defines the general types of components and allowable standards to be used and identifies any necessary guidelines for their application.

Logical Architecture

This is a design that meets a hypothetical set of requirements. It serves as a detailed example that illustrates the results of applying a generic architecture to specific circumstances.

Specific Architecture

The specific architecture addresses components, interfaces, standards, performance and cost. Specific architectures show how all the selected information security components and mechanisms combine to meet the security requirements of the systems under consideration.

Other architecture typologies exist, such as Figure 2.1 depicted over leaf. Conceptually this is the kind of typology used in this thesis.

The term “architecture” is used extensively in several fields of study and in diverse context, but generally refers to a conceptual, abstract or real, design or plan that describes the system, including the constituent parts and the relationships between them, its structure, organization, policies and standards. [4]

Architecture is the art of designing the human built environment. It is an interdisciplinary field, which draws on mathematics, science, art, technology, social sciences, politics, history, and philosophy [10]. “Architecture is a science, arising out of many other sciences, and adorned with much and varied learning: by the help of which a judgment is formed of those works which are the result of other arts.” (Marcus V. Pollio, died ca. 25 BC).

The diverse context of the term architecture ranges from the familiar use in the building sector to use in biology (e.g. to describe the architecture of the human anatomy composed of organs, the nervous systems and the circulatory systems), to use in

enterprise (e.g. architecture regarding the business structures, constituent processes and workflows) [8]. Architectures, which can be represented by diverse variety of abstractions, provide a guide to how a system may be realized.

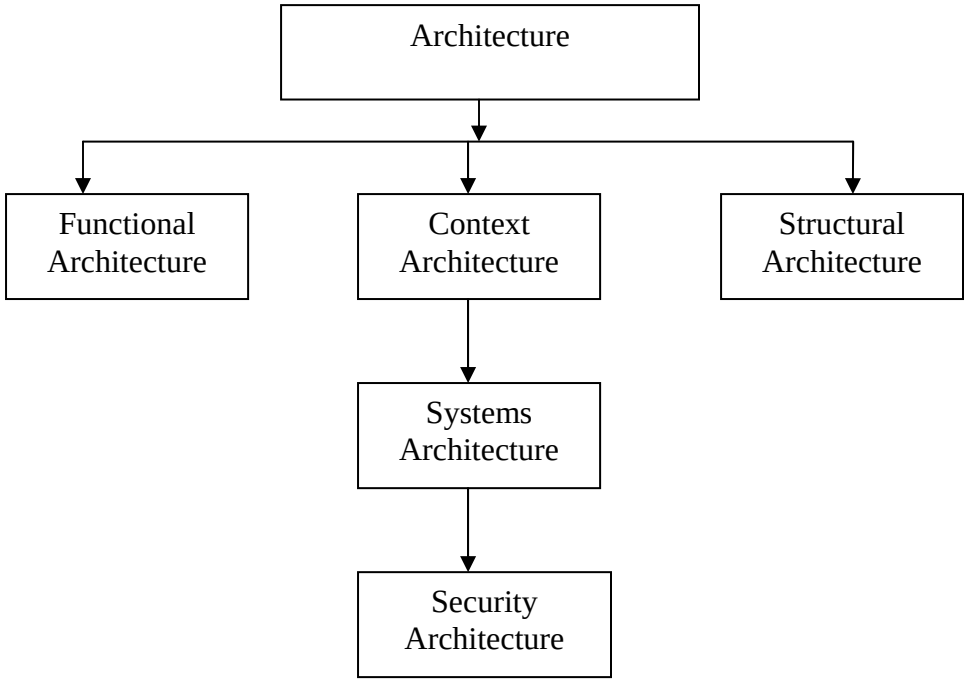

Figure 2.1: Typologies of Architecture

2.1.1. Systems Architecture. A system is any group of interdependent or temporally interacting parts or subsystems that come together to provide a service. Usually the parts or subsystems are systems themselves that provide service on their own and are generally composed of other parts. This generally implies that a system can be many things or comprise of many parts. This leads us to systems thinking which is a technique that may be used to study any kind of system be it human, natural, scientific or conceptual. The systems approach is based on the tenet that “the whole is more than the sum of the parts” — Aristotle (384 BC – 322 BC).

Fundamentally systems are a collection of different things which together produce results unattainable by the elements alone. Even though it may appear that a system can magically work together (since the subsystems already perform independently), it is

important to note that they have to be designed or architected in order to maximize their efficiency. Therefore, systems architecting focuses on the systems as a whole, particularly when making value judgments of what is required and design decisions of what are feasible [12].

Systems architecting is the art and science of developing systems solutions by using systems engineering specialties to develop satisfactory and feasible systems concepts and certification for client use in ill-structured problem environments [12].

Systems architecting approach goes beyond mathematical analysis and optimization of systems. Systems architecting quickly and naturally abstracts and generalizes lessons learned elsewhere, not only for itself but for transfer and specialization in other branches.

Systems architecting employs various tools among which are standards and heuristics. Heuristics are lessons learned, derived from experiences collated in several disciplines and over the years. Heuristics approach to security architecting is necessary because security is not static, but rather needs to evolve to keep up with changes.

Systems architecting is also characterized by these four attributes of performance-technical, aesthetic, sociopolitical, risk-uncertainty, complexity, management, cost-people, money, time and schedule-sequencing, events, coordination. All these require delicate balancing in order to achieve an optimal architecture.

2.1.2. Systems Architecting: the Context for Security Architecture. For this thesis, it is important to bring together all the elements of security to provide a systems approach methodology to security architecture. As system complexity increases, systems architects are faced with the increasingly difficult task of assuring that the evolving form of the system meets client needs. For security architecture, the systems tools of heuristics and standards can be applied, while also applying the mathematical analysis needed to achieve an optimal architecture.

Currently, security architecture requirements are largely specific to individual organizations; as a result a spread of definitions and understandings exists. In most, however, security architecture should provide a complete and consistent picture of security to allow for a step by step approach to managing security risks.

A security architecture should not only consider the structure of the technical components but many other facet such as complexity especially when it involves net-centric components , but it should also encompass the organizational and operational features such as the principles, policies, processes and their integration and interrelationships within the overall system it functions and contributes. This allows for an overarching business enterprise architecture, whether civil or military where consistency and compliancy is a requirement for success. Therefore Security Architecture [13] is an integral and critical component within the overall Enterprise Architecture designed specifically to:

- Enable secure communications and the appropriate protection of information resources within corporate infrastructures.
- Support legal information security requirements established by existing legislation pertaining to information confidentiality, accessibility, availability and integrity.
- Support secure, efficient transaction of business and delivery of services. As well as leverage opportunities to obtain IT Security synergies with the business.

2.1.3. Security Architecture for Net-centric Services Requirements. Security architecture services should address user identification, authentication, authorization & access control, administration and audit. The Defense Information Systems Agency (DISA) document [14] prescribes a security architecture for net-centric systems. The primary goal of the security architecture defined in this document is to ensure Enterprise Services (ES) can be invoked securely. As with every mission critical distributed system there is a set of key security requirements that must be met:

1. Authentication – Most (if not all) service providers will require that consumers are authenticated before accepting a service request. Service consumers will also need to authenticate service providers when a response is received. Different authentication mechanisms should be supported, and these mechanisms should be configurable and interchangeable according to service-specific requirements.

2. Authorization – In addition to authentication of a service consumer, access to a service will also require the consumer to possess certain privileges. These privileges feed an authorization check that is usually based on access control policies – who can access a service and under what conditions, for example. Different models may be used for

authorization, such as mandatory or role based access control. The authorization implementation should also be extensible to allow for domain- or communities of interest (COI)-specific customizations.

3. Confidentiality – Protect the underlying communication transport as well as messages or documents that are carried over the transport so that they cannot be made available to unauthorized parties. Sometimes only a fragment of the message or document (e.g. wrapped within a certain XML tag) may need to be kept confidential.

4. Data Integrity – Provide protection against unauthorized alteration of messages during transit.

5. Non-repudiation – Provide protection against false denial of involvement in a communication. Non-repudiation ensures that a sender cannot deny a message already sent, and a receiver cannot deny a message already received. This is especially important in monetary transactions and security auditing.

6. Manageability – The security architecture should also provide management capabilities for the above security functions. These may include, but are not limited to, credential management, user management, and access control policy management.

7. Accountability – This includes secure logging and auditing which is also required to support non-repudiation claims.

In addition other requirements necessary for a Service Oriented Architecture (SOA) mentioned include security trust domain, interoperability, security policies and so on.

2.2. STANDARDS AND HEURISTICS

Heuristics means “to guide” or “to pilot”, making it an important tool in architecting since architecting is a form of piloting. Heuristics [12] are abstraction of experience accumulated with a remarkable characteristic of it being passed on and used in the future. It helps avoid the pit falls of yesteryears and reduces it’s time consuming process. The formats of heuristics are words expresses in natural languages.

Heuristics provide non-analytical guidelines for treating complex, inherently unbounded, ill-structured problems. They are used as aids in decision making, value judgment and assessment. They are found throughout systems architecting, from earliest

conceptualization through diagnosis and operation. They provide the successive transition from qualitative, provisional needs to descriptive and prescriptive guidelines and, hence, to rational approaches and methods [12].

Heuristics helps in risk reduction as well as provides lessons to teach in the control of critical system features.

An example of a heuristics that applies to security architecting methodology is “The greatest leverage in systems architecting is at the interface”. With this heuristics attention is focused on the elements of security architecture and a structured methodology is prescribed through it.

Systems standards, another important tool set allows for system/ interface integration and interoperability. Examples of standards include the system specification, interface description and interface management.

The Joint Technical Architecture (JTA) is a Department of Defense (DoD) document [15] that provides standards, guidelines and specification for interoperability in communications and weapon systems. These sets of commercial specifications are provided in the areas of information processing, information transfer, modeling, message format, user interface, and security that need to be applied to all new information technology and national security systems.

In order to successfully overcome battlefield challenges, U.S. Forces will need to operate in a fast, flexible and agile manner. The necessary ingredient for this to happen is to provide quality information that can be shared without deterioration to enable sound individual and collective judgments.

The key to achieving this is to ensure the timely reception of secure and accurate data to the intended party. The intended party could range from the foot soldier of US force or another unit of the armed forces to coalition forces of allied countries, including aid agencies and other non-governmental organizations (NGOs).

Achieving this information end-state will result in forces attaining Information Superiority over potential adversaries. The JTA document came to exist because of the need for interoperability between the various components involved in the ever changing battlefield, including remote logistical and support operations.

Interoperability is achieved when systems can provide and accept services from other systems and can operate effectively together. Interoperability is defined by DoD as "The condition achieved among communications electronics systems or items of communications electronics equipment when information and services can be exchanged directly and satisfactorily between them and/or their users. The degree of interoperability should be defined when referring to specific cases"

JTA is intended as an open systems approach in designing weapons since it is not possible to determine beforehand all the systems and components that will be involved in any given battlefield.

The technical standards espoused by the JTA document is aimed at military network enabled systems. Other network enabled systems such as the banking systems also can benefit from standards and probably have in place technical standards specification for interoperability of equipment. This provides some sort of security measure for the system since any equipment that connects to the system must have the technical standards prescribed, however this is not sufficient in offering a consistent protection to the system in the event of a breach in security.

Both heuristics and standards are good tools in security architecture; however a methodology is also required.

2.3. ARCHITECTURAL FRAMEWORKS

The Open Group [16] is international consortium of vendor-neutral buyers and suppliers of technology. The main mission of this group is "to cause the development of a viable global information infrastructure that is ubiquitous, trusted, reliable, and easy –to-use. The Open Group creates an environment where all elements involved in technology development can cooperate to deliver less costly and more flexible IT solutions".

The Open Group Architecture Framework (TOGAF) evolved from the DoD's Technical Architecture Framework for Information Management (TAFIM).The Architecture Forum of The Open Group whose key activities includes defining, integrating and evolving standards to support Open systems is charged with developing TOGAF since inception and has evolved continuously since the mid-90's[16][17].

TOGAF is an Enterprise Architecture framework which provides a comprehensive approach to the design, planning, implementation, and governance of enterprise information architecture. This architecture is typically modeled at four levels or domains; Business, Application, Data, Technology.

Unlike the DoDAF [18], the architectural description dictates what products to assemble, further than the essential ones. Also TOGAF is iterative making it possible to rework processes until a better one is assured. This is typically what security architecture should be able to perform since it has been determined that security is dynamic. Figure 2.2 show that the phases navigate iteratively in a cycle. The circles represent the major phases of building and maintaining the enterprise architecture using the ADM.

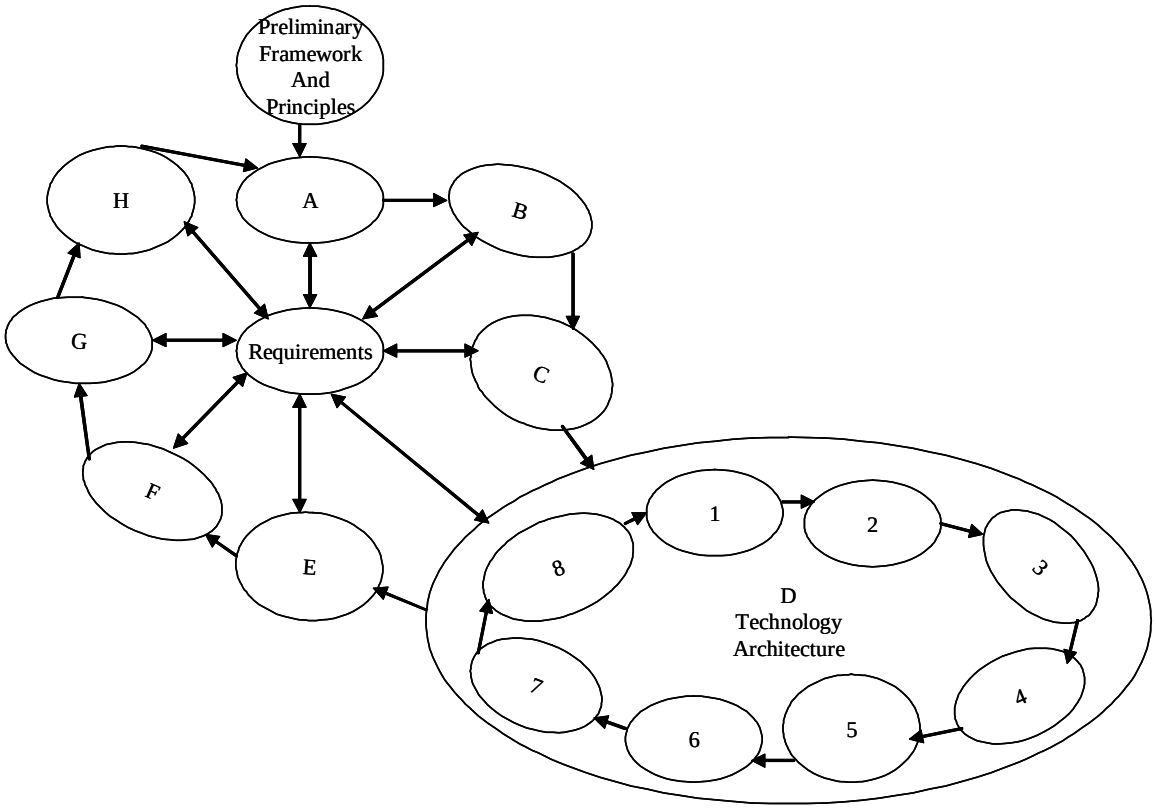

Figure 2.2: TOGAF Enterprise Framework. Source: The Open Group

Architectural frameworks also provide valuable tool for designing a security architecture methodology, particularly TOGAF provides for iteration which is an added value since security is dynamic and its methodology should also be dynamic to meet with security changes.

2.4. SECURITY ARCHITECTURE FOR NATO NEC

Melrose & Madahar proposes a Layered Structure for an overarching security architecture. In Figure 2.3 shown below they propose a framework for developing an overarching NATO Networked Enabled Capability (NNEC) Architecture using a previously performed NNEC Feasibility Study [19]. The framework consists of five abstraction layers. The dashed line provides a notational view of Networking and Information Infrastructure (NII). A NII reference architecture has been generated and includes consideration of Information Assurance (IA) elements but at the technical level. There is no dedicated security view within the architecture, or consideration of a business driven top-down approach, which would satisfy some of the reasons security architecture is necessary for Network Enabled Capability (NEC) mentioned previously {site Introduction Relevance}. With a focus just on NII, there is a risk that the security aspects of the architecture will be fragmented and not provide the required coherent enterprise-wide picture.

Even though the focus has largely been on logical security of data, physical security of equipment and infrastructures should not be overlooked. For the most part satellite and other resources such as copper wire and submarine cables has been the conduit for most of the network activities, albeit these resources are sometimes not located in close proximity for supervision rendering it a possible target for mischievous attackers. A very recent example can be seen in the destruction of an old satellite by the Chinese, reminding us all that physical security for these resources will need to be provided at some point.

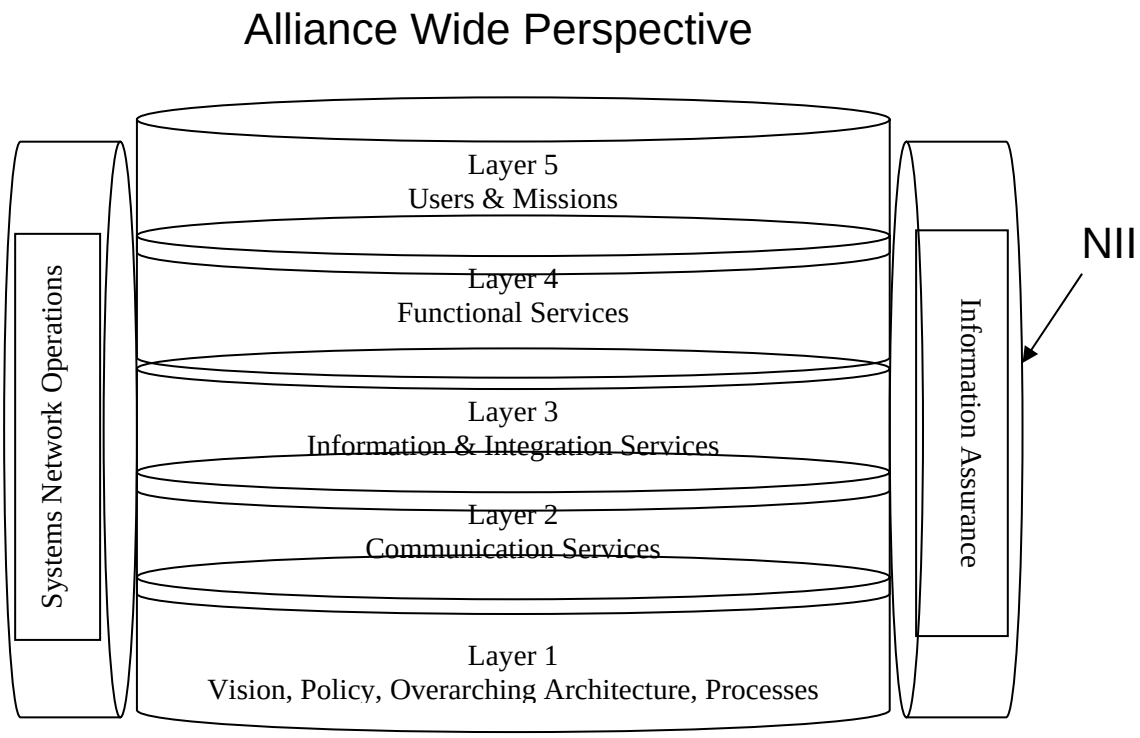

Figure 2.3: Framework for Development of an Overarching NNEC Architecture Proposed by Melrose & Madahar

3. METHODOLOGY

3.1. METHODOLOGY DEVELOPMENT

The methodology used in this thesis was to first describe in detail the different concept requirements of what security architecture should consist. By following the listed approach a methodology is developed, which is then applied in describing security architecture for small scale net-centric operation of the University of Missouri-Rolla.

3.1.1. Security Architecture Development. In systems engineering, security architecture means much more. Before delving into security architecture development, there is a need to throw more light on the term “interoperability” since several connotations can be drawn from different fields. Interoperability is the connecting of people, data and diverse systems. The Institute of Electrical and Electronics Engineers (IEEE) defines interoperability as the ability of two or more systems or components to exchange information and to use the information that has been exchanged [20].

According to International Organization for Standardization and the International Electrotechnical Commission ISO/IEC 2382-01, interoperability is defined as follows: “The capability to communicate, execute programs, or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units”[21].

Therefore, interoperability describes the capability of different programs, systems, units to exchange data, information or services through common sets of procedures in order to provide services. Interoperability strongly implies that the product or system be designed with standardization in mind.

Now because of all these connections going on, there is need to have an architecture framework that caters for security. Security architecture in this context embodies several concepts among which are:

(1) Providing both coherent and interoperability approach to security problem and solution.

(2) Describing the relationships between different parts of security solution.

(3) Providing a guide as to how security solution can be achieved.

Security architecture as a subset of architecture, specifically addresses elements relevant to security-relevant issues. The security architecture is a strategic framework that allows the alignment of an organizations development and operations effort. It is a unifying framework that allows reusability of services that implement policy, standards, and risk management decisions. In addition the security architecture allows for improvements which may not be possible to make at a project level. At the architecture level for instance, an architect can prospectively recognize the need to leverage a reusable service for several projects instead for only a particular project, thereby saving cost in the long run.

In summary, security architecture provides the framework and foundation to enable secure communication, protect organization business processes and information resources, and ensures that new methods for delivering service are secure.

3.1.2. Security Architecture Methodology. The words "method" and "methodology" indicate "a particular course of action." Literally methodology entails "the study of method". Methods impress order. Actually the idea of using method in a chaotic world is to glean the benefits of order it impresses [22].

The dictionary defines methodology as "a particular procedure or set of procedures". Checkland [23] writes "I take a methodology to be intermediate in status between a philosophy…and a technique or method. A philosophy… might be…'political action should aim at a redistribution of wealth in society,'…At the other extreme a technique is a precise specific programme of action which will produce a desired result: if you learn the appropriate technique and execute it adequately you can, with certainty, solve a pair of simultaneous equations…A methodology will lack the precision of a technique but will be a firmer guide to action than a philosophy. Where a technique tells you 'how' and a philosophy tells you 'what,' a methodology will contain elements of both 'what' and 'how'".

A methodology is defined as a codified set of practices, sometimes accompanied by training materials, formal educational programs, worksheets, and diagramming tools that can be carried out repeatedly to replicate a product or procedure. It can also be defined as an organized, documented set of procedures and guidelines for one or more

phases of the life cycle, such as analysis or design. Methodology usually includes methods, procedures, and techniques involved in analyzing information.

A security architecture methodology therefore refers to sets of practices for performing or defining security architecture in a coherent, consistent, accountable and repeatable manner. It presents a package of practical ideas, principles, procedures and proven practices that can be applied in the planning, design and development of security measures in any large or small scale system.

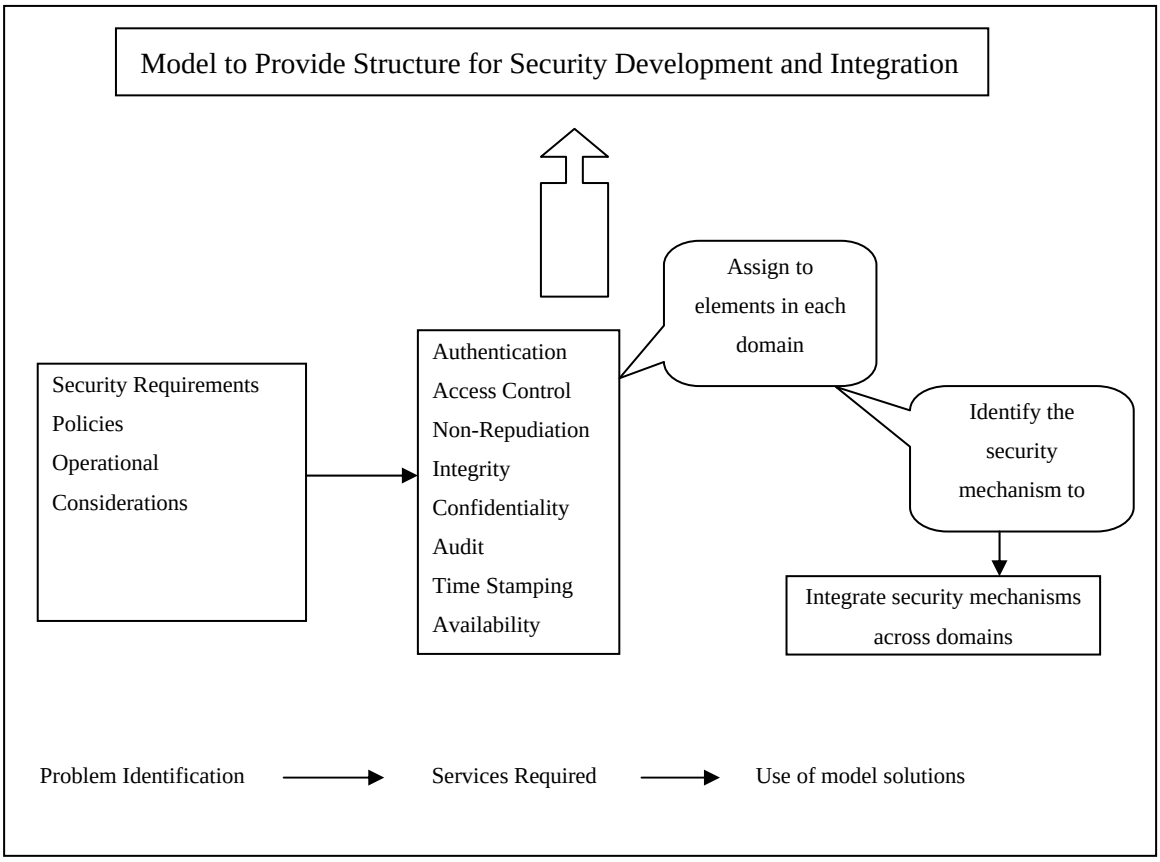

Figure 3.1: Architecture Development Methodology. Adapted from [9]

In designing a security architecture methodology, there are important terms that are necessary in outlining security. Such as security services, security process, security

management, security policy, risk management, security architecture and others. All these are elaborated subsequently.

3.1.3. Security Architecture Methodology Requirement. In designing any system methodology, some key requirements need to be met. Several methods of performing a requirements analysis for systems are available. However, each method has its strengths and weaknesses; therefore a combination of several methods was used in this requirements gathering to provide not only a broad range of usefulness, but represent different approaches to the problem of gathering requirements. Critical Success Factor (CSF) Analysis method and Usage Scenarios are the methods mostly used in this analysis.

3.1.4. Critical Success Factors Analysis. The concept of identifying and applying Critical Success Factors (CSFs) dates back to the original notion of "success factors" in D. Ronald Daniel's [24] "Management Information Crisis". Critical Success Factor (CSF) as it is known today was expanded by John F. Rockhart [25], of MIT's Sloan School of Management, from Daniel's work to specifically filter and identify the information in the making of critical enterprise decisions.

In "A Primer on Critical Success Factors," [26] Rockhart codified the ideology of success factors as a way to systematically identify the information needs of executives. It clearly specifies the essential steps of gathering and investigating data for the formation of a set of organizational CSFs that can be used by executives to aid in organizational administration. This document is generally thought to be the first account of the CSF method.

Even though both Rockhart and Daniel focused on refining the information needs of executives, Rockhart equally hinted at the value of the method as a component for the strategic planning of information systems or technology. Therefore CSF method has been used in many areas including in the technology planning methodologies in use today [27].

CSF used in this context is considered to be an essential component of a strategic plan that must be achieved in addition to the organization's goals and objectives. It is important to make this subtle distinction because an organization's CSFs should drive the accomplishment of its mission.

CSFs identify key performance areas that are crucial in order for an organization or process to achieve its mission. Administrators or overseers of a project should know and consider these key areas when they set goals, as they provide a common point of reference for the entire organization or process, when they direct operational activities and tasks that are important to achieving goals,. Thus, any activity or initiative that the organization undertakes must ensure consistently high performance in these key areas; otherwise, the organization may not be able to achieve its goals and consequently may fail to accomplish its mission [27]. Therefore a good security architecture methodology should outline key performance areas that should support strong security architecture.

Traditionally, strategic planning and managements' definition of a goal or an objective is moderately well known; conversely, characterizing CSF is not particularly clear. Hence, CSFs are often confused with organizational goals. In this paper organizational goals are defined as targets that are established to achieve the organization's mission. They are very specific as to what must be achieved, when it is to be achieved, and by whom. Effective goals have a quantitative element that is measurable to determine if the goal has been achieved. Goals can be decomposed into operational activities to be performed throughout the organization.

CFS can be defined in many ways, these points to the elusive nature of CFS. The Critical Success Factors analysis methodology is a top-down approach for determining requirements based on the needs of the organization. The top-down approach makes CSF a perfect tool, well-suited for determining requirements analysis for large systems with many stakeholders and audiences with various interests and at times conflicting. CSFs are those key or important things that need to be realized in order to achieve the goals. A CSF for an organization or process in general is usually related to more than one goal. Some of the merits of identifying CSFs are that;

- They are simple to understand and help direct awareness to major concerns.
- They are good method of communicating to workers, implementers and can be easy to monitor.
- They can be used in concert with strategic planning methodologies.

Identifying CSFs is extremely important because it keeps people focused and each CSF should be measurable and associated with a target goal because things that are

measured get done more often. Exact measures are not necessarily important, but they provide guidelines.

The CSF analysis method starts off by determining the goal of the mission, commonly termed mission statement, and then goes on to separate high-level goals of the mission statement.

The high-level goals are then decomposed into Critical Success Factors, which are then split into many levels of hierarchy, becoming more specific.

At the lowest level, each CSF becomes a requirement for the system; a single, well-defined task that must be accomplished in order to be successful. Along the way, problems to be solved and assumptions made are recorded.

Once the CSF hierarchy is established and a set of requirements has been derived, these can then be arranged into a matrix for comparison with the problems identified. In order to be considered complete, each problem must be fully addressed by one or more requirements.

By analyzing the steps necessary to achieve success, and cross-referencing them against problems to be solved, a complete set of requirements can be determined that can then be correlated with specific user scenarios. Each of the requirements should apply to at least one user scenario, and, generally, more than one.

This methodology allows requirements to be determined that satisfy the needs of the organization and those of the user. Since architectural frameworks are built and maintained by organizations, this method allows us to create a well-defined and reasonably complete set of requirements.

3.2. THE ANALYSIS HIERARCHY

3.2.1. Mission. The mission of the Security Architecture methodology is to develop and maintain standard reference for designing and implementing any large scale network-centric system.

It is important to note that the primary users of this document include stakeholders, therefore the need to map the system’s stakeholders’ conceptual goals to a logical view for security managers who need to make decisions on security, as well as other audiences such as the information technology community and other

managers/implementers who are developing security architectures for use as a reference architecture.

3.2.2. Goals. At the highest level, the goals of this security architecture methodology can be divided into 5 categories. Each of which is related to the CSFs and requirements which will be explained, where each of the top-level goal is further elaborated.

It is also important to take cognizance of the fact that this security architecture methodology covers the needs of enterprises that engage in net-centric activities, which rely heavily on the information technology.

The Top-level Goals for the Security Architecture and its methodology are;

Reliability: The security architecture must be reliable and stable over time. Critical success factor and requirement for this goal is enabling the security services to be reliable, stable and evolvable over time.

Integratability: The security architecture must be consistent with current and future needs of a security.

Scalability and Extensibility: The security architecture must enable implementations that are scalable and extensible.

Team Goals: The security architecture should meet the needs of the user community. Critical success factor and requirement for this goal include; it should be reliable, stable and evolve over time and it should be consistent and coherent

Management and Provisioning: The standard reference security architecture should provide for manageable, accountable environment for security. Critical success factor and requirement for this goal should be to enable the management and provisioning for security.

3.2.3. Critical Success Factors for Security Architecture Methodology. The list below shows the critical success factors needed for security architecture methodology.

(1) It should provide guidelines for security policy & standards.

(2) It should provide guidelines for security management.

(3) It should provide risk assessment and management.

(4) Account for security services.

(5) It should include security federation.

(6) Provide for security auditing.

3.3. SECURITY REQUIREMENTS

Security requirements identify types and levels of protection necessary for equipment, data, information, applications, and facilities. Below are identified some key requirements that must be met for a secure system. These requirements apply to a large extent to information technology security, although it also applies to other environments that require security. The objectives of security requirement are;

- Ensure that users and client applications are identified and that their identities are properly verified.
- Ensure that users and client applications can only access data and services for which they have been properly authorized.
- Detect attempted intrusions by unauthorized persons and client applications.
- Ensure that unauthorized malicious programs (e.g., viruses) do not infect the application or component.
- Ensure that communications and data are not intentionally corrupted.
- Ensure that parties to interactions with the application or component cannot later repudiate those interactions.
- Ensure that confidential communications and data are kept private.
- Enable security personnel to audit the status and usage of the security mechanisms.
- Ensure that applications and centers survive attack, possibly in degraded mode.
- Ensure that centers and their components and personnel are protected against destruction, damage, theft, or surreptitious replacement (e.g., due to vandalism, sabotage, or terrorism).
- Ensure that system maintenance does not unintentionally disrupt the security mechanisms of the application, component, or center.

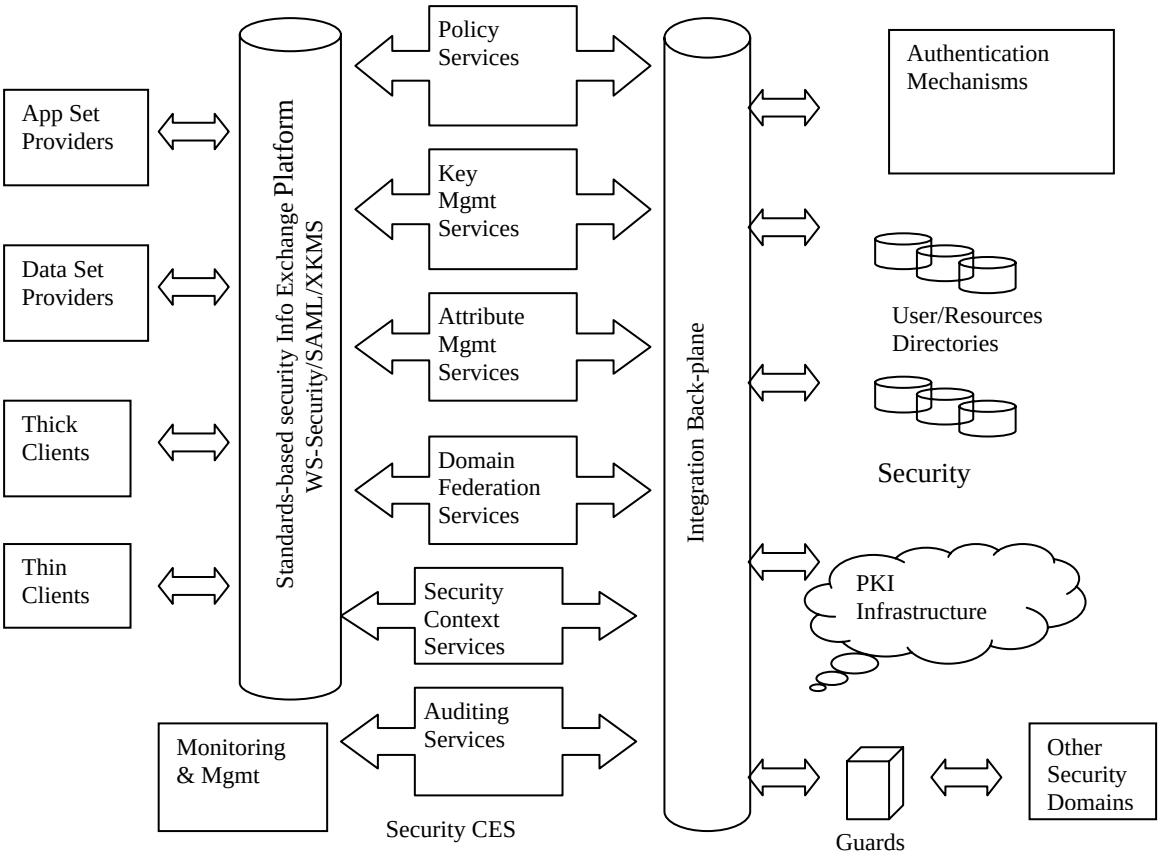

Figure 3.2: Conceptual Enterprise Security Architecture. Adapted from [14]

3.3.1. Security Policy and Standards. Security policy refers to organizational guidelines and principles that govern the system's design, operation, and run time. The security policy describes what is permissible in a system as well as what the system cannot permit. Security standards should be prescriptive guidance for designing and operating systems, and should be backed by reusable services wherever practical. Security should not be regarded exclusively as an intermediary, but requires an architecture and design advocate and backing at runtime.

Security policy and standards are not end goals in themselves, they need to be backed by a governance model (federation) that ensures they are in use, and that it is practically possible to build, deploy, and operate systems based on their intent. In practice

this means that the security architecture must define reusable security services that allow developers to not be security experts yet still build a secure system.

A security policy should not determine how a business operates; the nature of the business should dictate the policy. Defining a company's security policy can seem difficult, but by defining policy before choosing security methods, organizations can avoid having to redesign security methodologies after they are implemented.

Security policies should map to an organization's business objectives, regulatory issues and industry best practices enhancing their ability to implement strong information security and avoid legal and regulatory liabilities. Security experts should work closely with each organization to determine their unique business needs objectives, environments and cultures. In addition, expertise and thorough understanding of current and future regulatory, industry trends and globally accepted standards should be crafted into a policy as it becomes available; this will lead to greater security, regulatory compliance and enhanced business practices. Features of a balanced security policy are listed below.

- Provides a comprehensive set of protection criteria based on the availability, confidentiality and integrity requirements of the organization, in other words a course of action should be provided.
- Defines roles and responsibilities appropriate to an organization in support of the protection policy developed, in other words a guiding principle based on corporate policy.
- Access to market-leading risk management and industry-best practices expertise.
- Assists an organization in protection policy implementation, acceptance and awareness.
- Builds an enterprise policy framework that is manageable today and in the future.
- Procedures that are considered expedient, prudent, advantageous, and productive

  <u>Benefits of a security policy</u>

  (a) Fast availability of a customized security policy.

  (b) Provides an objective, expert assessment of current security policies.

  (c) Provides the basis for establishing a security awareness program.

  (d) Supports consensus building for security policy implementation.

(e) Ensures compliance with security regulations and limits access to confidential data.

(f) Lowers insurance premiums by reducing risks associated with adverse security issues.

3.3.1.1 Effective security policies. Effective security policies should consider the factors discussed in the following sections.

1. Identifying Resources

Clearly identifying organizations resources will focus attention to potential adversaries who might want to undermine the organization, therefore organizations must know what they want to protect, what access is needed, and how these consideration work together. Security measures usually do not prevent unauthorized users from trying to break security systems; they can only make it more difficult, therefore companies should make a decision on what to focus on, the value of their assets or otherwise.

2. Cost Implications

Some security measures might inevitably diminish expediency, particularly for advanced users. This might lead to the delay of work and may create costly overheads. When instituting security policies, companies should weigh cost against the potential benefits. Depending on the cost-benefit analysis, some infrastructures might be left unprotected, if they do not have costly implications when compromised.

3. Identifying Assumptions

Making assumptions provides a starting point when designing a policy. For example, an organization might assume that any user is savvy enough to break any security code if they are dedicated. It is important to examine and justify assumptions; any hidden assumption is a potential security hole.

4. Controlling Secrets

Passwords and encryption keys should be kept secrets. Most importantly, areas to be protected need to be kept secret. Knowledge with which an organization’s security can be circumvented should be guarded carefully to ensure that adversaries don’t get their hands on it. The more secrets there are, the harder it will be to keep all of them. Security systems should be designed so that only a limited number of secrets need to be kept.

5. Appreciate the Environment

It is important to comprehend the functions of the system in the environment and understand how each unit contributes to the system. This will help in identifying abnormal behaviors and enable the setting of the proper security policy.

6. Physical Security

While logical security is important, physically securing an organization's resources is equally important. Security policy on who is granted physical access to any resources should be emphasized.

7. Providing Pervasive Security

When changes are made to the system, the security policy should be upgraded to encompass the change. This is especially true when new services are created. System administrators, programmers, and users should consider the security implications of every change they make. Understanding the security implications of a change takes practice; it requires lateral thinking and a willingness to explore every way that a service could potentially be manipulated. The goal of good security design and policy is to create an environment that is not susceptible to every minor change.

Security policies are living documents, because organizations are constantly subject to change, security policies must be systematically updated to reflect new business directions, technological changes, and resource allocations.

3.3.1.2 Methodology for prescribing security policy and standards. One approach to setting security policies and procedures is suggested by the following steps:

(a) Identify all the assets that need to be protected.

(b) Identify all the vulnerabilities and threats and the likelihood of the threats occurring.

(c) Look at what policy is currently available; if none exist develop one by gathering information from various sources and applying it specifically to organizational needs

(c) Decide which measures which will protect the assets in a cost-effective manner.

(d) Communicate findings and results to the appropriate parties.

(e) Upgrade current policy and standards with most recent updates and make sure to send out the people charged with implementation.

(f) Implement the policy following the established guidelines. Example obtaining user ID, verify user ID etc.

(g) Monitor and review the process continuously for improvement.

3.3.2. Security Risk Management. Any security architecture methodology should take into consideration risk management, as an important organizing concept. Risk is comprised of assets, threats, vulnerabilities, and countermeasures. A risk management centric approach allows for the security architecture to be agile in responding to security needs. Risk is a function of threats exploiting vulnerabilities against assets. The threats and vulnerabilities may be mitigated by deploying countermeasures. The risk management process implements risk assessment to ensure the systems' risk exposure is in line with risk tolerance goals. This does not mean that behavior is uniformly risk averse or risk seeking. The system should take on the appropriate level of risk based on the set goals.

By building in risk management, any security risk that is breeched can be undertaken by designing and deploying countermeasures that allow for sensible security risk.

Risk management does not eliminate risks entirely, however its' role in the security architecture is to educate people about the risks they are taking and provide countermeasures in the event that the undertaken risk does not suit the defined goals.

The dynamics encountered by organizations require that organizations sometimes have to make decision on whether to embark on vulnerability-based or risk-based approaches to security management. A risk-based approach has become the norm, which features cost prominently, thereby allowing proper cost associations to be placed on systems. The equation below properly defines the components involved in assessing risk.

$$Risk = \left(\frac{Threats \times Vu\ln erabilities}{Countermeasures} \right) \times Assets$$

Figure 3.3: Risk Equation

Risk-based approaches have been used successfully in diverse areas such as marine operations, building construction, financing and engineering to improve various odds. Uncertainty is the main characteristics of risk.

The ability to utilize risk management successfully lies in the practical solutions for dealing with this uncertainty. Documentations showing reduction in negative risks that is attributed to the application of risk management abounds.

The emergence of risk management in the last fifty years as an interdisciplinary field of study known as decision science is aimed at a formalized method to improve risk reduction. This analytical approach which was initially applied to aircraft safety and nuclear power has rapidly spread to other applications.

Risk management is a continuous process that assesses/mitigates risks. Assessment is activity of identifying and analyzing risk. Probability (or likelihood) is chance that risk will occur. Consequence is unfavorable result of risk. Mitigation is action taken to lower probability and/or consequence of risk. Risk Level is numerical or qualitative assessment of risk based on risk's occurring probability and consequence.

Risk management constitute an iterative process involving five important step, which are planning, identification, assessment, analysis and handling.

Several risk assessment methods exists, such as the R. von Solms' [28] traditional assessment vs. baseline control and the quantitative vs. qualitative bifurcation, all have pros and cons. The Sandia report [22] discusses risk assessment method that is divided into three archetypical approaches identified as "temporal", "functional", and "comparative", corresponding to stress testing, threat analysis and lifestyle, respectively.

However, in this paper attack trees was chosen in order to approach risk assessment as a risk-based rather than as vulnerability-based.

3.3.2.1 Risk analysis utilizing attack trees. Risk analysis is a method used in determining ways to in which risk can be eliminated or minimized. Possible solutions are accomplished by utilizing trade-off studies and other analytical methods.

The term security really does not have a lot of meaning unless it is feasible to determine how long security needs to be provided or who to provide security for. A good method of going about securing a system is to model security threats against it. This method provides insight and understanding to the diverse ways in which a system can be

attacked, so that countermeasures can be designed to foil those attacks. Also understanding the means, motivations, and objectives of the attackers will go a long way in helping to design better countermeasures to foil attempts by the attackers.

3.3.2.2 Attack trees and GSN. Attack trees [29] present a method for evaluating the security of systems. They provide a method for the capture and reuse of the expertise gained in security, and the ability to respond to changes in security. Since security is a continuous process, attack trees provide a basis of understanding that process.

Attack trees make use of the Boolean logic to determine what needs to happen before an attack on a node can occur, originating from one node and propagating through the other nodes in the system. This is represented using a tree structure; with the goal designated as the root node and the diverse methods of getting to the goal being designated as the leaf nodes. The combination of attack trees and Goal Structuring Notation (GSN) [29] technique provides better safety assurance for a system.

GSN is used in safety-critical industries to improve the structure, rigor, and precision of safety arguments [29]. Figure 3.4 overleaf shows the combination of the two techniques. It basically consists of three levels, the top most level is the Goal, the next is the Strategy and the last level is the Solution. The Strategy level usually has sub goals.

From the structure it can be deduced that each level is successively decomposed into the next level until a point is reached where nodes can be supported by direct reference to available solutions [30]. The argument in diagram above can be summed up with the “IF’ statement below.

IF {(Solution 1 ***OR*** Solution 2) ***OR/AND*** (Solution 1 ***AND*** Solution 2 ***AND*** Solution 3)] } ***THEN*** Top Goal

In employing the attack tree model, value/cost is placed on each node. Another method could be rating the vulnerability of each node. In the GSN attack tree [30] model, each node is rated based on the risk assessment determined using a scale of one through five where 1 corresponds to very low, 2 = low, 3 = moderate, 4 = high, and 5 = very high. Analysis of the risk factor known as the Value At Risk (VAR), decisions on the procedure to be undertaken can be made depending on the vulnerability.

In determining the VAR, the attacker's motive plays an important role, since it influences the rating of the particular node. The VAR is set high if the attacker's aim is very important.

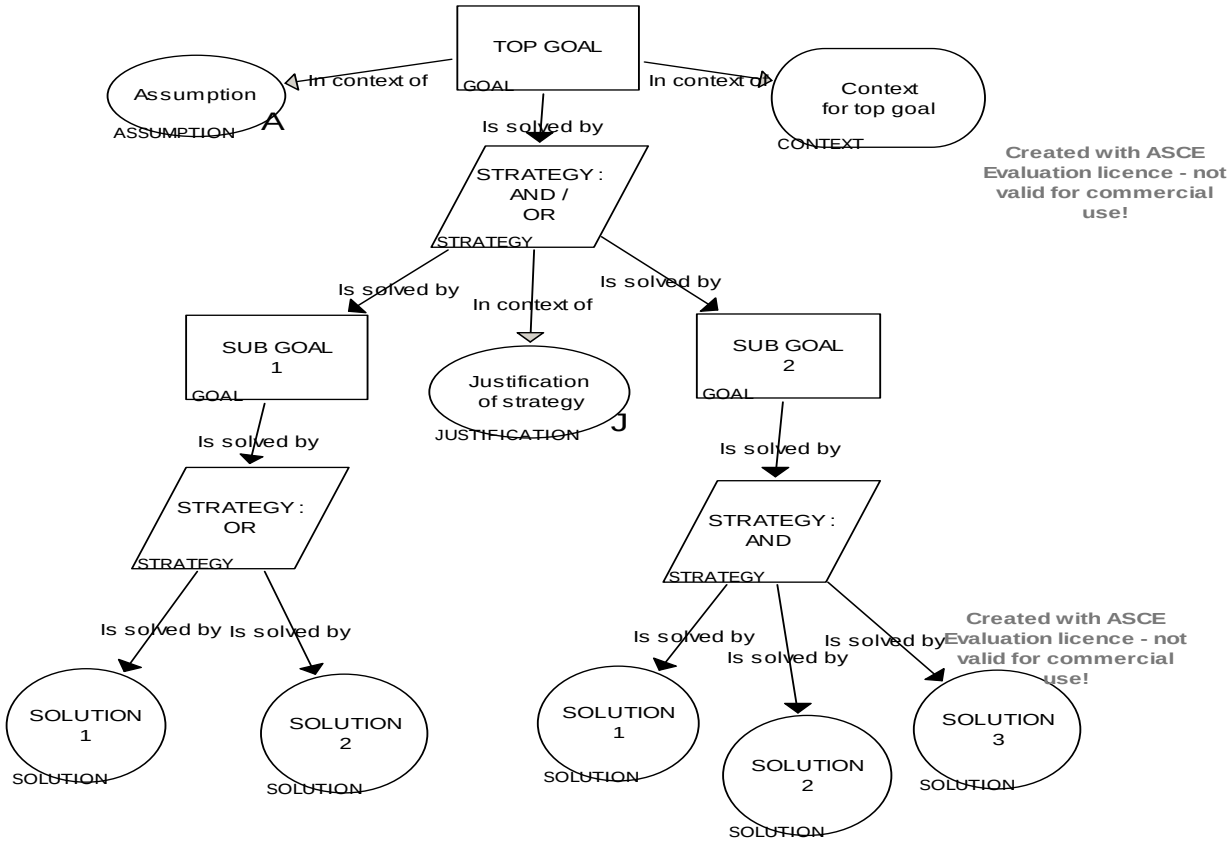

Figure 3.4: Structure of an Attack Tree using GSN

3.3.2.3 Countermeasures, traceability and recovery mechanisms. In order to deploy countermeasures on a compromised system, the security policy (which should be robust and resilient) which has been developed for the system needs to be applied.

An Assurance Case arguing the security of critical system attributes contributes significantly towards evaluating the VAR [31]. The NERC homepage has the guidelines and best practices in detail, for developing an acceptable security policy [32].

When and if an attack occurs, it is important to trace its source so that proper measures can be deployed to counter such attacks in the future. The VAR should be set higher if a trace is not possible due to its seriousness.

A prompt assessment of any damage to the system should be done and recovery preformed using the damage recovery technique provided. Also the VAR on the compromised node should be set to a lower value.

3.3.2.4 Methodology for prescribing security risk management. The itemized below describes steps necessary to provide for security risk management.

(a) Identify risks

(b) Analyze risks

(c) Prioritize risks

(d) Define avoidance and alternate for each risk

(e) Define mitigation plan

(f) Define contingency plan

(g) Implement plan, track risk and revise risk management strategy.

3.3.3. Security Auditing. Auditing which usually means to formally conduct an examination of vital components of an organization sometimes brings to memory images of unending witch hunting in some organizations. However conducting an audit enables an organization to take a second look at what they are doing thereby enabling them to make better decisions.

Security audits are assessments of how the confidentiality, availability and integrity of an organization's system are assured. Conducting a security audit is one of the superb methods of determining whether an organization's security measures are effective or not without incurring the cost and other associated damages of a security incident.

Security auditing specifies the extent to which a business, application, component, or center shall enable security personnel to audit the status and use of its security mechanisms.

Security audit is also a systematic, measurable technical assessment of how the organization's security policy is employed at a specific site. It involves providing independent evaluations of an organization’s policies, procedures, standards, measures, and practices for safeguarding any information both electronic and otherwise from loss,

damage, unintended disclosure, or denial of availability. The broadest scope of work includes the assessment of general and application controls.

Security auditors work with the full knowledge of the organization, at times with considerable inside information, in order to understand the resources to be audited. Security auditors work by conducting personal interviews, vulnerability scans, examination of the organization's system settings, analyses of networks, and historical data.

Security audits are part of an on-going process of defining and maintaining effective security policies. It should involve constant iteration using the feedback derived from the processes. It involves everyone who uses any security related resources throughout the organization. Security audits provide the tool that enables a fair and measurable way to examine how secure a site really is.

The current state of technology requires audit steps that relate to testing controls of access paths resulting from the connectivity of local-area networks, wide-area networks, intranet, Internet, etc., in the IT environment.

3.3.3.1 Objectives of security auditing. Typical objectives of security auditing are to ensure the collection, analyzing, and reporting of information about the:

- Status (e.g., enabled vs. disabled, updated versions) of its security mechanisms.
- Use of its security mechanisms (e.g., access and modification by security personnel).

An example of security auditing is an application that can collect, organize, summarize, and regularly report the status of its security mechanisms Identification, Authentication, Authorization, Immunity, Privacy, and Intrusion Detection. Security mechanism for security auditing should be implemented using mechanisms like audit trails and event logs.

The results of these evaluations are generally directed to the organization's management, legislative bodies, or other auditors. Information security auditing may be performed in engagements where

- The specific audit objective is to evaluate security, or
- The audit objectives are much broader, but evaluating security is a necessary subset. (For example, an audit objective such as financial statement assurance or

program evaluation frequently may be met only when there is assurance that the security of the financial or program data is adequate.)

3.3.3.2 Methodology for prescribing security auditing. The itemized below describes steps necessary to provide for security auditing.

(a) Congregate an audit team from the accounts and IT departments.

(b) Define the scope of audit by creating assets list and security perimeter.

(c) Create a threat list, examine threat history and check security trends & past audits. (d) Review current policies. (e) Perform a survey of the site. (f) Use questionnaires where necessary. (g) Develop audit plan/checklist.

(h) Meet with site managers to determine what data will be collected, how/when will it be collected, site employee involvement, and answer questions

(i) Report findings and update security policy.

3.3.4. Security Domain Federation. Network security environments consist of dissimilar or diverse constituents. The political concept of "federation" takes on new meaning due to these diverse constituents. The arrangement in which no one group or organization manages all users and resources in a distributed application environment is describes as federation. In this scenario, administrators in diverse domains enact local security policies that support the mutual benefits of transactions among their respective area of operation.

The word federation originates from the Latin word trust hence its tie to the trust domain. In distributed network services, federation refers to the need for trust agreements among decentralized security and policy domains.

Federation allows access-management functions to span sundry organizations, business units, sites, platforms, products and applications. Federation necessitates that an organization trust other site administrators to validate its own users' identities. A federated environment, allows users to log on and access resources transparently in external domains that are subject to various policies defined by both internal and external administrators.

Users can log on through authentication techniques either through an ID/password or Kerberos, and this authentication is communicated to a federated destination site through an authentication assertion. Kerberos is a computer network authentication

protocol, which allows individuals communicating over an insecure network to prove their identity to one another in a secure manner. It requires a trusted third party.

Federation provides the mechanisms for cooperation between different interworking network domains possibly owned by different administrators. In order to be able to offer services to their users, these administrators must cooperate.

Security federation specifically refers to an approach that requires a centralized focus on security issues. Federated security enables collaboration across multiple systems, networks, and organizations in different trust realms. It is a mechanism that allows for clean separation between a service and its associated authentication and authorization procedures for clients consuming the service. Three key elements of federated security architecture are:

Domain/Realm: This can be a single unit of security administration or trust. A typical domain might include a single organization.

Federation: A collection of domains that have established trust. The level of trust may vary, but typically includes authentication and almost always includes authorization. A typical federation might include a number of organizations that have established trust for shared access to a set of resources.

Security Token Service: A Web service that issues security tokens; that is, makes assertions based on evidence that it trusts, to whoever trusts it. This forms the basis of trust brokering between domains. The Figure 3.5 below illustrates an example of federated security.

In this scenario, there are two organizations: A and B. Organization B has a Web resource (a Web service), that is of some value to users in organization A.

3.3.4.1 Trust domain. Trust is important both in people's daily lives and in networks, because it gives peace of mind to the system administrator that unauthorized users can be kept away. Quoting Sen. Chuck Hagel, R-Neb., "... We govern with one currency, and that's trust. And that trust is all important. And when you lose or debase that currency, then you can't govern……..."

Interoperability and connectivity needs of equipment in a net-centric environment raise trust issues, and trust management becomes pertinent. Trust is essential in distributed and net-centric systems because of the need to allow resource sharing,

concurrent processing and operations. Communications in these systems traverse domains and organizations, and same trust level differ for each of the domains. Yet, within the same domain, users' trustworthiness can diverge.

Parties involved in the trust include but not limited to the different services of the armed forces, the platforms, equipment, personnel, logistics support, contractors that

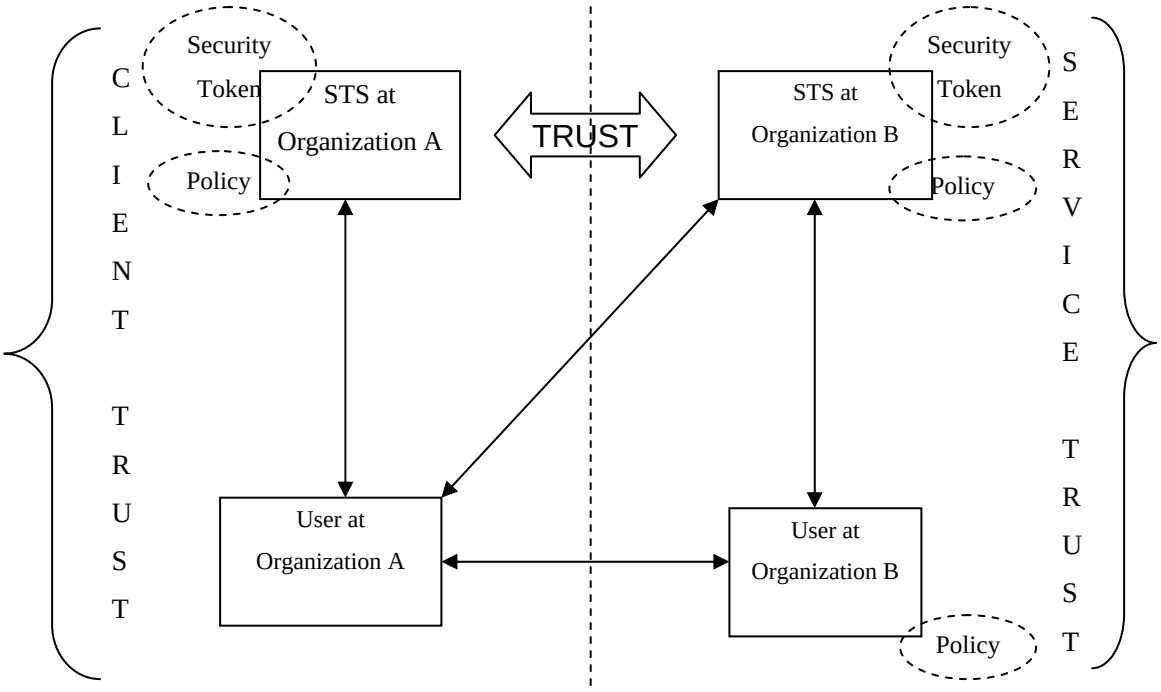

Figure 3.5: Trust Relationship between Two Organizations.

currently rely or intends to rely on networked environment enhanced by advances in information technology.

Various dictionaries define trust as an assured or firm reliance on character, integrity, ability, strength or truthfulness of a person or something. Another definition of trust is the belief that an entity is capable of acting reliably, dependably and securely in a particular case. [33].

Trust management on the other hand is a "unified approach to specifying and interpreting security policies, credentials, and relationships that allows direct

authorization of security-critical actions." [34]. Trust management is the collecting of all the necessary information used in establishing trust in a relationship, while continuing in the monitoring and adjustment of existing trust relationships[35].

A flexible and general-purpose trust management system can help maintain current and consistent credibility information for the different entities in a net-centric system.

A trust management system unites the concept of security policy specification with the method for specifying security credentials. Credentials describe specific delegations of trust. Trust management credentials relate directly to authorizations to perform specific tasks. Trust management systems support delegation, and policy specification and refinement at the different layers of a policy hierarchy, thereby implementing consistency and scalability. In addition, trust-management systems are by design extensible and can express policies for different types of applications.

The trust management approach, initiated by Blaze et al. [34] requires that "the set C of credentials prove that the request r complies with the local security policy P". Each entity that receives requests must have a policy that serves as the ultimate source of authority in the local environment. The policy may directly authorize certain actions to be taken, but more typically it will delegate this responsibility to credential issuers that it is certain to have the required domain expertise as well as relationships with potential requesters.

The trust-management engine is a separate system component that takes (r, C, P) as input, outputs a decision about whether compliance with policy has been proven, and may also output some additional information about how to proceed if it has not been met.

Proofs of compliance can be determined by the use of a general purpose, application-independent algorithm which is an important part of a trust-management approach. This is a good idea since any product or service that requires some form of proof that compliance with policies has been met could use a special purpose algorithm implemented from scratch. The advantage of using a general-purpose compliance regulator lay in its soundness and reliability of both the definition and the implementation of "proof of compliance."

A compliance regulator of a general-purpose nature can be explained, formalized, proven correct, and implemented in a standard package, and answers returned for any given input (r,C, P) depends only on the input and not on any implicit policy decisions in the design or implementation of the compliance regulator. However, in order to design this sort of a trust-management engine some ground rules need to be provided. They are:

– A definition of “proof of compliance”

– Policies and credentials should be fully defined

–Responsibility should be assigned between the trust-management engine and the calling application. For example the application may obtain all credentials needed for the compliance proof before the trust management engine is invoked, or the trust-management engine may obtain additional credentials while it is still constructing a proof.

Distributed systems imply that the systems are located at different places possibly in time and space. It is a collection of independent computers/systems that appears to its users as a single coherent system. In order for these systems to work together they need to communicate through the use of networking resources. Because they are not physically accessible, there has to be some way of verifying that indeed it is the right equipment when they try to make contact with other distributed systems, therefore the need for trust models.

Need for Trust Management

The list below [35] describes the reasons for trust requirements for security in a distributed system.

Authentication: The identity of users in a distributed system is often not well known. Hence the need for some form of authentication to be performed before the decision to grant access can be made. On average, authentication is achieved through a username/password mechanism. Straightforward password-based protocols are insufficient in networked computing environments, however, even against unsophisticated adversaries; simple eavesdropping can destroy security. Other mechanisms include:

• One-Time passwords, which do not secure the rest of the session.

• Centralized ticket-based systems, such as Kerberos [36]. Problems with such systems include the necessity for an authentication server (and for frequent communication with it) and implicit trust assumptions.

• Public-key based authentication protocols, which are considered the “state of the art” for scalable authentication systems.

Delegation: Scalability in distributed system is dependent on delegation. It enables decentralization of administrative tasks. Existing distributed-system security mechanisms usually delegate directly to a “certified entity.” In such systems, policy (or authorizations) may only be specified at the last step in the delegation chain (the entity enforcing policy). The implication is that high-level administrative authorities cannot directly specify overall security policy; rather, all they can do is “certify” lower-level authorities. This authorization structure leads easily to inconsistencies among locally-specified sub-policies.

Local trust policy: The number of administrative entities in a distributed system can be quite large. Each of these entities may have a different trust model for different users and other entities. For example, system A may trust system B to authenticate its users correctly, but not system C; on the other hand, system B may trust system C. It follows that the security mechanism should enforce uniform and implicit policies and trust relations.

A trust model for distributed systems can be illustrated using various trust models, such as public key cryptography, the resurrecting duckling model, and the distributed trust model. These models will be briefly discussed and its relationship to security will be explored.

3.3.4.2 Trust models. Below are some trust models used currently in peer-to-peer systems and other networks.

Public Key Cryptography or Simple Public Key Infrastructure (SPKI) was one of the early standards proposed for distributed trust management. It implicitly utilizes trust management concept by identifying and authenticating parties seeking to establish contact.

Resurrecting duckling model is a hierarchical structure with a master-slave relationship. The mother duck is described as the master entity while the duckling is the

slave entity. Instruction or the secret key is passed from the master entity through a secret channel to the slave entity. A slave entity can in turn become a master entity when it passes a secret key called imprinting to another entity.

Distributed trust model is based on "conditional transitivity of trust" [33] which simply put means that trust is transitive under some condition. [37] This model relates to the human society where trust is generated by both direct and indirect interactions. That is, entities can obtain information and recommendations from other sources other than the main source. However because recommendations have uncertainty or risk, entities need to know how to cope.

Two types of distributed trust model exist based on asymmetry; they are direct trust and recommender trust. Trust relationship is grouped between the two entities in terms of different interactions. Trust in one group does not depend on trust in the other group. The model utilizes continuous trust values for direct trust and recommender trust, depicted in the tables shown below.

Table 3.1. Direct Trust Value

Value	Meaning	Explanation
-1	Distrust	Completely untrustworthy
0	Ignorance	Can't decide
1	Minimal	Lowest trust
2	Average	Mean trustworthiness
3	Good	Trusted by major population
4	Complete	Fully trustworthy

Table 3.2. Recommender Trust Value

Value	Meaning	Explanation
-1	Distrust	Completely untrustworthy
0	Ignorance	Can't decide
1	Minimal	The entity itself judges the reliability of recommender's recommendation
2	Average	
3	Good	
4	Complete	

The recommendation protocol is straightforward. For example, entity A needs a service from entity D (say joint coalition force). Now, A knows nothing about the trustworthiness of D's service, so A asks B for a recommendation with respect to the possibility of joint coalition in the near future, assuming A trusts B's recommendation within this category. When B receives this request and finds that it doesn't know D either, B forwards A's request to C, which has D's trustworthiness information within joint operations. C sends a reply to A with D's trust value. The path A _ B _ C _ D is said to be the recommendation path.

The following formula is used to calculate the trust value from the returned value1: tv_T = [rtv(1)/4] _ [rtv(2)/4] _ ... _ [rtv(i)/4] _ ... _ [rtv(n)/4] ? tv(T), where rtv(i) is the trust value of the ith recommender in the recommendation path, tv(T) is the trust value of target T returned by the last recommender, and tv_T is the calculated trust value of target T. When multiple recommendation paths exist between the requester and the target, the target's eventual trust value is the average of the values calculated from different paths.

This model exhibits some weaknesses discussed below:

- The model does not consider false recommendations and assumes that a recommender with a good recommender trust value always makes reliable recommendations, which might not be true.
- The model does not provide a mechanism for monitoring and reevaluating trust, which is dynamic.

Binary concept of true or false should not be used in trust management since trust is relative. Abdul-Rahman and Hailes[37] quantify trust as a multiple value concept. Several other trust management systems apply similar method. To curtail the possibility of the wrong trust recommendations, the trust model is grouped in two

- *Evidence-based model*, here entities create trust relationships based on some previous evidence, such as keys [38] [39] [40];
- *Recommendation-based model*, here recommendations from intermediaries establish the trust relationship between two strangers. Trust management systems for distributed systems can be placed into these two categories.

3.3.4.3 Recommendation-based trust management. The previously mentioned study conducted by Xiong and Liu on trust management in distributed peer-to-peer systems was based on feedback or recommendations of unknown or unfamiliar peers. While P2P systems involves entities that may not know each other from Adam, they entities that will use the JTA most likely know each other but may not have worked tighter. Thus, a slight twist to the equation for this study. However, the recommendation-based trust model can be applied to this study when some of entities involved may have worked together previously.

Xiong and Liu define a satisfactory interaction as 1 and a complaint as 0 in their trust metric below. Applying this equation to this gives the following:

$$T(u,t) = \frac{\sum_{v \in P, v \neq u} S(u,v,t).Cr(v,t)}{\sum_{v \in P, v \neq u} I(u,v,t)} \quad (1)$$

where

• P is a set of entities in the system;

• u and v are entities in the system, $u, v \_ P$;

• $S(u,v,t)$ is the degree of satisfaction that u has with v until the tth transaction;

• $T(u,t)$ is u's trust value evaluated by other entities until the tth transaction;

• $Cr(v,t)$ is the balance factor for filtering feedback from v; and

• $I(u,v,t)$ is the number of interactions that u has with v up to the tth transaction.

Therefore, T(u,t) is the ratio of the cumulative weighted satisfaction that u receives to the total number of interactions that u has within the system.

S(u,v,t) _ Cr(v,t) by I(u,v,t) – C(u,v,t) _T(v,t). C(u,v,t) is an approximation which shows the degree of complaint filed by v against u. and C(u,v,t) _ T(v,t) indicates the filtered complaint filed by v against u. T(u,t) becomes:

$$T(u,t) = 1 - \frac{\sum_{v \in P, v \neq u} C(u,v,t).T(v,t)}{\sum_{v \in P, v \neq u} I(u,v,t)} \quad (2)$$

T(u,t) falls within the range of (0, 1) as prescribed by Xiong and Liu . The higher T(u,t) is, the more trustworthy u is. This approach uses v's trust value T(v,t) as a balance factor. The higher T(v,t) is, the more reliable v's complaint is. Thus v's complaint has more impact on u's trust value. The trustworthiness decision criterion is:

If I(u,t) > C1 and T(u,t) > C2, then u is trustworthy.

C1 and C2 are thresholds, with C1 defining the minimum number of interactions required. Obviously, a certain number of interactions are necessary to improve accuracy. Xiong and Liu's approach considered both positive and negative evaluations and interaction history, and therefore more likely to produce accurate results. Some drawback associated with this trust management system is explained below:

- A minimum number of interactions are required in equation two's decision criteria, therefore a possible disadvantage for an entity which has not worked with any one of the group in the system.
- Equation one uses a balance factor which is a trust value assigned by one the entities the system assumes that an entity with a higher trust value always gives more reliable feedback than an entity with a lower trust value, which might not be true.
- There may come a time when the entity's behavior may change due to prevailing circumstance. The most recent feedback will be closer to the entity's current behavior than older feedback; however this model utilized same weight in evaluating an entity's trust.

A similar trust management system utilizes feedback to evaluate trust value; however, it only considers complaints making the system too sensitive to misbehavior. Decision is made by using probabilistic method to analyze complaints.

Security policy and standards normally dictates standards that all equipment interacting with its' system should adopt. This can be translated as a sign to promote trust among genuine systems wishing to utilize the capabilities of a system in a networked environment. However, there is a need to use the trust model and trust management concept.

3.3.5. Security Management. [27] Relates that managing security is one of the many problems that confronts organizations and must be resolved in order for its mission to be accomplished. Regardless of the assets that need securing—information or technical assets, physical plant, or personnel—a security strategy that can be deployed, measured, and modified as becomes necessary is a must have for the organization. The effectiveness of any security strategy depends on how well it fits with the CSFs as well as the organizations' other missions.

When problems traverse an organization, it poses many management challenges, especially when it concerns security. First and foremost the areas of import should be identified and targeted. This requires the organization to take an inventory to determine what needs to be protected and why. In a large, complex organization, this can result in the identification of hundreds of assets that are important to strategic drivers. Next, in order to secure these assets special skills and resources will be required, typically scattered throughout the organization. Because security is a concern for the whole organization, its' management is no longer seen as a sole proprietary of information technology department.

Security issues are becoming increasingly complex, and the need for a single, centralized point of management is becoming increasingly necessary. As the threat in the environment grows, compliance issues are making it essential to secure an ever-increasing perimeter.

A centralized philosophy of management needs to be adopted, backed by a robust security infrastructure with immediate Event Management responsiveness, and backed by Information Management long-term configuration and log analysis support.

Security management is further discussed by looking at two equally important subfunctions, security services and security processes.

3.3.5.1 Security services. Security services provide confidentiality, integrity, and availability services for a platform. Security services are implemented as protection services, such as authentication and authorization, detection services, such as monitoring and auditing, and response services, such as incident response and forensics. These services have served as the goals and objectives for information security programs for many years, but they do not provide an actionable blueprint as such. Later on in this

document, a method to map these security services into an overall security architecture plan will be described.

Security services can also be described as those services that support access control on objects and non-repudiation of operations on objects. Access control is defined as a Security service that gives admittance to a user. Non-repudiation, which is also a security service, provides proof that an action was carried out by a particular user at a particular time.

3.3.5.2 Security services and strengths required. The security services and strengths required for network protection are discussed below.

Authentication: Authentication of all personnel in this domain is required.

Access Control: Access to data objects in this domain shall be granted on an individual basis. Access control shall restrict functions available for all data sets on example documents. Individuals filing research documents shall have read access only to data sets pertaining to pending actions. Government personnel processing the documents shall have read access to all data sets in this domain and read/write access to their assigned documents.

Data Integrity: Data integrity shall be provided for all information within the domain.

Confidentiality: Not mandatory for document filings but the capability is required as a choice should the filer so require. Required for all document processing done by the Government after receipt.

Non-Repudiation: Not required for document filing but the capability is required as a choice. Required for all Government initiated actions pertaining to pending document actions. Individuals must be positively identified and time stamping for electronic document filing actions is required.

Audit: All access to the systems within the domain to modify data objects shall be audited.

Availability: Since DOCT is a research and development system, specific availability mechanisms will not be implemented.

Attribute Services

Various attributes are needed to support policy-based decisions. These attributes are those of the principals, the system resources, and the application environment. This service group provides standard access mechanisms for such attributes, and defines how attribute queries are returned.

Principal Attribute Service – provides query and retrieval interfaces to access attributes for principals, which may be individuals or even organizations. The attribute taxonomy or "schema" is not defined by the service, but rather by the underlying attribute authorities (e.g. identity stores). These attributes are retrieved and provided upon request and may be used as inputs to the policy decision logic

Other attribute services are Resource Attribute Service for retrieving resources, Environment Attribute Service for retrieving environment attributes and Attribute Administration services to actively manage the attributes.

Credential Services

Credential service provides identification and recommendation that enables subscribers participate in electronic transactions. It provides access to the underlying security infrastructure. If a credential service provider offers more than one type of credential then each one is considered a separate credential service. Some of the services included in credential service include:

- Certificate Validation Service (CVS) – CVS makes it possible for clients to assign part or all certificate validation responsibilities. This is particularly important for clients who do not have the capability for Public Key Processing (PKI).
- Certificate Registration Service – Public Key certificates are necessary to utilize digital signatures and encryption. Assuming that clients generate their own public/private key pair, their equivalent certificates need to be generated, hence the need for a certificate registration service. This services the required protocol that enables the use of the public/primary key system.
- Certificate Retrieval Service (CRS) – CRS helps provides authentication verification, digital signature verification, and public key encryption operations for users and clients alike.

3.3.5.3 Security process. Risk management, security policy and standards, and security architecture govern the security processes and defense in depth architecture through design guidance, runtime support, and assurance services. Security metrics are used for decision support for risk management, security policy and standards, and security architecture. The security architecture should have a reference implementation for developers and other IT staff to review what functions the security mechanisms performs, and how they do it.

Security processes carry out the intent of the system risk management, security policy and standards, and security architecture. They are broken into discrete domains because they solve very different problems, and require different staffing, support models, and success criteria.

3.4. OVERARCHING METHODOLOGY FOR SECURITY ARCHITECTURE

(1) Create security policy & standards by

(a) Identify all the assets that need to be protected.

(b) Identify all the vulnerabilities and threats and the likelihood of the threats occurring.

(c) Look at what is currently available; if none exist develop one gathering information from various sources and applying it specifically to organizational needs

(c) Decide which measures which will protect the assets in a cost-effective manner.

(d) Communicate findings and results to the appropriate parties.

(e) Upgrade current policy and standards with most recent updates and make sure to send out the people charged with implementation.

(f) Implement the policy following the established guidelines. Example obtaining user ID, verify user ID etc.

(g) Monitoring and review the process continuously for improvement.

(h) Adopt technical standards where necessary

(i) Ensure that users comply by providing access to information on ways to procure and install/implement.

(2) Establish security federation for the organization by

(a) Designing trust models

(b)Devise procedures for establishing trust.

(3) Establish a risk management approach including

(a) Identify risks (b) Analyze risks (c) Prioritize risks (d) Define avoidance and an alternate for each risk (e) Define mitigation plan (f) Define contingency plan (g) Implement plan (h) Track risk (i) Revise risk management

(4) Incorporate security auditing process

(5) Security management process is necessary to ensure that all the security services and processes are co-coordinated properly.

Figure 3.6 below depicts the methodology diagram consisting of the five elements discussed and their relationships.

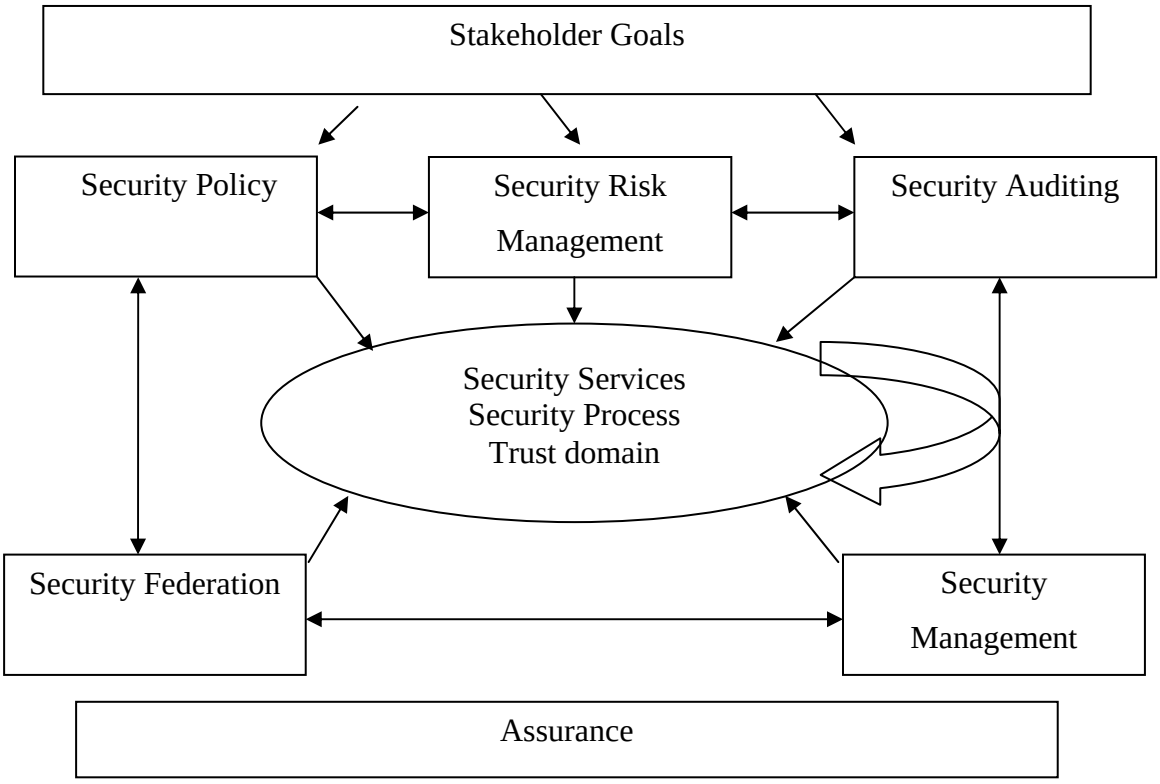

Figure 3.6: Security Architecture Methodology Process.

4. THE UNIVERSITY OF MISSOURI-ROLLA SYSTEM

The University of Missouri-Rolla (UMR) is one of four campuses in the University of Missouri System. The university is a research school that offers educational programs in major engineering and other scientific disciplines that are technology-based, technology-dependent, or which support these programs. UMR has a student population of over 5000 each semester in about 2,160,199 gross sq.ft. acreage. Classes are held at the main campus in Rolla and at several off campus locations. Also the advances afforded by information technology have made it possible for students to enroll and attend classes virtually from all parts of the globe; nearly 100 classes are taught via distance education in each of the fall and spring semesters. This trend has ensured that UMR is among the higher education institutions with net-centric capability. This, in turn, has presented the UMR IT department some challenges, one of which is designing a security architecture for its net-centric system.

There is no one single solution for a security architecture for a net-centric capability in a higher education institution such as UMR, but using the common elements of security architecture for NEC developed in the previous sections, a suitable plan of action can be developed for UMR. This security architecture needs to be reviewed periodically and updated as needed or as a result of security audit reports.

4.1. CHALLENGES FACED BY UMR IT SECURITY

Universities and other higher education institutions such as UMR have networks that are frequently open, to facilitate collaboration between students, faculties and research organizations that are not on the campus. This also means that computers from off campus sites which connect to the university's network might already be compromised, making it a lot easier for viruses, worms, and other malicious software to spread throughout the network. Therefore both outside (off campus) and inside (on-campus) threat issues will need to be considered.

Also open networks which allow computers outside the campus to connect to the campus networks are more susceptible to hackers than computers in corporate networks as depicted by a test at the University of Maryland, Baltimore County (UMBC) and

similar test at other universities which show that security can be compromised in a matter of hours. T This is because the open, collaborative network environments at universities are seen as easy targets by the hackers [41].

Another challenge is that UMR has over 6500 students and faculty, who may have little or no training in good system administration practices, each possibly using individual computers/laptops.

Because IT infrastructures are constantly evolving, it therefore means that a security architecture can never be complete and will need revising at intervals. On the other hand, this constant change can be advantageous since it may present opportunities to leverage new technologies.

Viruses are written to achieve the most possible damage and reach the widest possible audience; in most institutions, Microsoft operating system and Microsoft software suite are used, therefore devising a security architecture that puts focus on this will be of immense advantage.

4.2. PROBLEM DEFINITION

The problem faced by UMR can be viewed from two perspectives: network protection and information protection. As stated in the challenges above, a major problem for UMR is how best to secure the information in an open network. How best to assure users that the information going through the network is secure without infringing on the rights and expectations for an open network. Another requirement of similar importance is how to protect the network from attacks that might cripple the network.

In order to tackle this problem, a first step is to assess the mission statement, which includes both the use policy and the e-mail policy.

4.3. REASONS TO IMPLEMENT SECURITY ARCHITECTURE

(1) Information Protection: to assure that the information on students, staff, and faculty is not accessed by unauthorized persons.

(2) Intellectual Property: to ensure that only authorized students/faculty/staff can access course materials.

(3) Integrity: to ensure the integrity of data stored in accounts.

(4) Access Control: To protect computers inside the perimeter protected by traditional firewalls.

4.4. CURRENT SECURITY APPROACH AT UMR

Currently, there is no specific approach or single document utilized by UMR, rather the security policy and activities are scattered throughout different areas. One example of a security approach in place is to give several privilege levels to students and others. Subsequent paragraphs compares UMR security architecture with the five security architecture elements developed in this thesis.

UMR appears to have three levels of policy and standards: an IT policy, a UMR policy and an overarching UM System policy. The policy and standards are good first steps; however, they can and should be improved. One very positive step that UMR has taken over the past few years is the appointment of a full-time Information Security Officer. There is also an IT Coordination Committee which reviews progress and policies.

UMR IT performs a qualitative risk assessment by performing a daily threat analysis. They also use an Educause tool, which is vulnerability analysis software for residual risk examination. They have seen a 10%-19% increase in improvement in their ability to detect threat.

UMR IT does not use external auditors to audit the network; rather they keep a main log and monitor accounts periodically. The entire University of Missouri System conducted an internal audit several years ago which led to more consistent policies across all four campuses. At UMR, students accounts are usually left for 6-12 months after the students graduate; employee accounts are frozen immediately upon leaving UMR and all privileged are revoked.

For federation, they rely wholly on recommendation-based trust in allowing network users access. There is a plan in the works to use shibboleth in the future for a federated identity based authentication and authorization. Shibboleth will allow for information about users in one security domain to be provided to other organizations in a common federation, which will provide cross-domain single sign-on identification (SSID) and will remove the need for content providers to maintain usernames and

passwords/passphrases. Identity providers supply user information, while service providers consume this information and gate access to secure content.

In the management area, UMR has a business impact analysis assessment process which consists of business continuity and disaster recovery techniques, recovery time objectives, reliability goals, multiple feeds to pick up slack time (that is, the utilize redundancy so that no time is lost in the event of a recovery) and assets recovery (which is not yet available).

4.5. USERS OF THE NETWORK

There are diverse usages of the UMR network and users' privileges vary. Some users have only a basic privilege, while others have very high level privileges. The network use by student ranges from entertainment to research; therefore they are given the basic of privileges.

The next usage level is for educational use by faculty and research professionals and they have a higher level privilege. There are research users whose privilege ranges from low to extremely critical. There is also business usage by administrators within the UM System office; they have high to critical privilege. The network is also used for distance education where there are several privilege ranges. There are also staff users.

There are external users of the network, one of which is the United States Geological Survey (USGS). USGS uses it for rapid access mapping for rescue operations. All these users require different trust and privilege levels.

4.6. IDENTIFYING RESOURCES TO PROTECT

Human Resources Office: This office collects and maintains information on all persons employed by the university, including their social security number which can easily be used by identity thieves to ruin a person's credit. Therefore protection of all computers in this office is a must and should reside at a high level.

Accounts Department: This office is equally important because it keeps records of all account information of students. If a computer in this office is compromised it can make it impossible to determine who has paid fees and who has not.

Email Server: Emails are so prolific today that younger generations must wander how older generations thrived without it. Likewise its usage on campus are enormous. Users are entitled some degree of assurance that their emails are sent only to the intended party and nowhere else.

Research Computers: Research is the bedrock of most institutions and successful research is critical to institutions such as UMR. Therefore it is important that research computers are not compromised whether to destroy or steal information and intellectual property.

Other Computers: Other computers in the system also need to be protected since some of them can log in to some of the other mentioned computers.

Network: The network is the conduit through which all these computers communicate. Constant monitoring of the network is necessary to locate lapses and intruders and to identify malicious activity of insiders within the network.

4.7. IMPLEMENTATION PROCESS

There are five elements of a security architecture as developed in this thesis; therefore the focus is on the elements of most importance for UMR with its open network. It is important to note that all five elements are necessary for successful security architecture; however some elements will be stressed more than others. Given the resources identified above as deserving protection, the categories into which they fall are examined closely, starting with security risk management and ending with security policy & standards.

4.7.1. Security Risk Management. Security risk management is important as already highlighted, more so for an open network such as ones found in UMR. Therefore more work can be done in this area in order to ensure the integrity of the network. Using the attack tree methodology as described in this thesis would enable and promote a rigorous security risk management plan for UMR network.

4.7.2. Security Audits. Even though logs are maintained and accounts monitored periodically, there is a need for an external auditor who will provide an objective approach different from what is obtained internally. External auditors can provide

unbiased audits that will enable the discovery of vulnerabilities that may lead to the revision of the security policy and standards.

4.7.3. Security Federation. The future use of the shibboleth system and any other trust model will surely be beneficial to the UMR network. The use of the network by the USGS might raise the trust level; therefore, in addition to the recommendation-based trust; the use of other trust models mentioned in this thesis will be needed in order to have a good grasp of the federation required.

4.7.4. Security Management: Security Services & Security Process. The UMR IT Office already utilizes various techniques in security management, one of which is the assignment of various privilege levels to categories of users. In the future, as they implement the shibboleth program, there will be a need for attribute and credential services. There will also be a need to improve access control to restrict access to some network sites by the anticipated users from other colleges and institutions. In addition, a documentation of the procedure on Event Management responsiveness, backed by Information Management long-term configuration and log analysis support, will go a long way in providing a starting point for security officers in the future.

4.7.5. Security Policy and Standards. The security policy's hierarchy is thus UM System: UMR: IT [42]. These policies and procedures are quite extensive. As more improvements are made to the network, it will to be updated, especially after audits are completed.

4.8. OVERARCHING METHODOLOGY

- Gather information about network security strategy, technology, policy, and devices
- Analyze network security architecture and design
- Identify, confirm, and reduce vulnerabilities in network security architecture, devices, and features
- Document security risk analysis and provide recommendations
- Provide an onsite presentation of findings and prioritized recommendations.
- Revise security policy and standard where needed.

5. CONCLUSIONS AND FUTURE WORK

This research proposed a security architecture for network enabled systems using five elements and further prescribed a security architecture methodology by leveraging these five elements. The security architecture while similar to [14] draws from several sources with an added methodology for developing each element. Using mathematical analysis and heuristics as tools, each element was further developed to show approaches of describing them. The challenges of prescribing a security architecture for net-centric systems were enumerated and discussed at length. The relevance of developing a security methodology for large scale net-centric operations, despite the obstacles, can be deduced since society has come to rely on such networks for daily activities and more.

An overview of system architecture was presented to show conceptually where a security architecture resides in relation to systems architecting and architecture in general. Also security architecture requirements for net-centric system were offered. The methodology began by carefully performing critical success factors analysis bearing in mind the goals of the proposed security architecture methodology.

Furthermore, the five elements of security architectures, namely Security Policy & Standards, Security Risk Management, Security Auditing, Security Federation and Security Management of services and process, were individually elaborated and developed by presenting mathematical analysis where applicable and methodologies were in turn prescribed. An overarching methodology was aggregated and put forward.

The methodology can be scaled to size for smaller net-centric operations such as UMR, by focusing on the important elements. Because of the open nature of an educational institution’s networks, focus can be shifted to the elements of most importance or where a security breach is most likely to occur.

Following the methodology developed, the current security architecture at UMR was presented and recommendations given for the future.

Future work in this area would be to implement this security architecture by following the procedures demonstrated in this thesis, as well as upgrading this methodology since it has already been established that security is dynamic entity.

APPENDIX

DEFENSE INFORMATION SYSTEMS AGENCY (DISA), MARCH 1, 2004. A SECURITY ARCHITECTURE FOR NET-CENTRIC ENTERPRISE SERVICES VERSION 0.3.

For the reference document please visit:

http://horizontalfusion.dtic.mil/docs/specs/20040310_NCES_Security_Arc.pdf

BIBLIOGRAPHY

1. Umeh, N., Miller, A., and Dagli, C.,"TOGAF vs. DoDAF: Architecting Frameworks for Net-centric Systems," Proceedings of the 2007 Industrial Engineering Research Conference, G. Bayraksan, W. Lin, Y. Son, and R. Wysk, eds.

2. Bruce Schneier - *Secrets & Lies, Digital security in a Networked World*. John Wiley & Sons (2000).

3. ISO/TC97/SC 16, "Reference model of open systems interconnection," Doc. N227, June 1979.

4. Melrose, J., and Madahar, B., 2007 "Security Architecture for NATO NEC."

5. Rechtin, E., Systems Architecting, Creating & Building Complex Systems, Prentice-Hall, Englewood Cliffs, NJ, 1991.

6. Edmonds, B., "What is Complexity? – The Philosophy of Complexity per se with application to some examples in evolution."

7. (Suh, yıl). Suh, N.P., "A Theory of Complexity, Periodicity and the Design Axioms."

8. NATO Network Enabled Capability Feasibility Study Version 2.0 dated June 2005.

9. Department of Defense Goal Security Architecture (DGSA) Transition Plan. Version 1.0, Jan. 30, 1995.

10. Articles concerning Modernism and Postmodernism, retrieved on April 16,2006 http://astudio.id.or.id/artkhus6modernpostmodern.htm.

11. Dynamic Architecture: How to Make Enterprise Architecture a Success, Martin van den Berg et al, ISBN: 0-471-68272-1, 256 pages, January 2005.

12. Maier, W., Rechtin, E., The Art of Systems Architecting 2nd ed. CRC Press, Boca Raton, FL, 2002.

13. Secure Shapes: Security Architecture, retrieved July 2007 http://secureshapes.com/secure_shapes_services/security_architecture.html.

14. Defense Information Systems Agency (DISA), March 1, 2004 "A Security Architecture for Net-centric Enterprise Services Version 0.3."

15. Department of Defense Joint Technical Architecture vol. II, version 6.0, October 3, 2003.

16. TOGAF, Dec. 2003, (The Open Group Architecture Framework) Version 8.1 Enterprise Edition http://www.opengroup.org/architecture/togaf8-doc/arch/.

17. David Harrison and Lou Varveris, 2004, "TOGAF: Establishing Itself As the Definitive Method for Building Enterprise Architectures in the Commercial World."

18. Department of Defense Architectural Framework Vol. III Version 1.5, Feb. 28, 2007.

19. Lowman, T., & Mosier, D., "Applying The DoD Goal Security Architecture as a Methodology for the Development of System and Enterprise Security Architectures," 1997 IEEE.

20. Institute of Electrical and Electronics Engineers. IEEE Standard Computer Dictionary: A Compilation of IEEE Standard Computer Glossaries. New York, NY: 1990.

21. ISO/IEC 2382-01, Information Technology Vocabulary, Fundamental Terms.

22. Phillip L. Campbell, Jason E. Stamp, "A classification Scheme for Risk Assessment Methods," SANDIA REPORT SAND2004-4233, August 2004.

23. Checkland, P., "Systems Thinking, Systems Practice," (1981) Wiley [rev 1999 ed].

24. Daniel, D. Ronald, "Management Information Crisis," Harvard Business Review, Vol. 39, No. 5, September/October (1961), III.

25. Rockhart, John F. "Chief Executives Define Their Own Data Needs." *Harvard Business Review 57*, 2 (March-April 1979).

26. Rockhart, John F. & Bullen, Christine V. "*A Primer on Critical Success Factors."* Cambridge, MA: Center for Information Systems Research, Massachusetts Institute of Technology, 1981.

27. Richard A. Caralli, "The Critical Success Factor Method: Establishing a Foundation for Enterprise Security Management," July 2004.

28. R. von Solms, "Can Security Baselines replace Risk Analysis?" IFIP TC 11 Conference on Information Security, Research and Business, 14-16 May 1997, Copenhagen, Denmark. pp. 91-98.

29. Tim Kelly and Rob Weaver; University of York - *The Goal Structuring Notation –A Safety Argument Notation - DSN 2004 Workshop on Assurance Cases* (July 2004, Florence, Italy).

30. Moleyar, K., and Miller, A., "Formalizing Attack Trees for a SCADA System," Department of Electrical and Computer Engineering, University of Missouri – Rolla.

31. Robin E. Bloomfield, Sofia Guerra, Ann Miller, Marcelo Masera, and Charles B. Weinstock - *International Working Group on Assurance Cases (for Security)* – IEEE Security & Privacy (May 2006).

32. North American Electric Reliability Council (NERC) – *NERC CIP cyber security Standards* www.nerc.org.

33. Huaizhi Li and M. Singhal, "Trust Management in Distributed Systems," IEEE Computer, Vol 40. No 2, February 2007, pp. 45-53.

34. M. Blaze, J. Feigenbaum, and J. Lacy., "Decentralized Trust Management." In *Proc. of the 17th Symposium on Security and Privacy*, pages 164–173. IEEE Computer Society Press, Los Alamitos, 1996.

35. Matt Blaze , Joan Feigenbaum , John Ioannidis , Angelos D. Keromytis, "The role of trust management in distributed systems security," Secure Internet programming: security issues for mobile and distributed objects, Springer-Verlag, London, 2001.

36. S. P. Miller, B. C. Neuman, J. I. Schiller, and J. H. Saltzer. Kerberos authentication and authorization system. Technical report, MIT, December 1987.

37. A. Abdul-Rahman and S. Hailes, "A Distributed Trust Model," *Proc. New Security Paradigms Workshop*, ACM, Press, 1997, pp. XX-YY.

38. M. Elkins, D. Del Torto, R. Levien, T. Roessler, *Mime Security with OpenPGP*, IETF RFC 3156, Aug. 2001; www.ietf.org/rfc/rfc3156.txt.

39. R. Housley, W. Ford, W. Polk, D. Solo, *Internet X.509 Public Key Infrastructure, Certificate and CRL Profile*, IETF RFC 2459, Jan. 1999; www.ietf.org/rfc/rfc2459.txt.

40. F. Stajano and R.J. Anderson, "The Resurrecting Duckling: Security Issues for Ad Hoc Wireless Networks," *Proc. 7th Security Protocols Workshop*, LNCS 1796, Springer-Verlag, 1999, pp. 172-194.

41. Suess, J., “Computer and Network Security in Higher Education.” John Wiley & Sons (2003).

42. University of Missouri-Rolla Security Policy, Retrieved on July 2007 http://www.umr.edu/it/policy/index.html.

VITA

Njideka was born in Ebonyi state, Nigeria on October 8th 1974. Being strongly inclined towards engineering, Njideka pursed a Bachelor's of Engineering in Electrical/Electronic Engineering (Communications Engineering option) at the Federal University of Technology-Owerri, Imo State, Nigeria. She worked as an intern in the three options of Electrical/Electronic – Power systems, Computer and Communications engineering. Upon receiving of her BS in August 1999, she worked for General Telecoms Plc as a pupil engineer and was promoted to Communications Engineer after the mandatory probationary period. After seeing firsthand challenges faced by engineers in dealing with management issues, Njideka developed interest in engineering management and systems engineering. Her continued interest in these areas prompted her to purse a degree in Systems Engineering at the University of Missouri-Rolla in Fall 2005, after completing an earlier degree in Summer 2004 in Information Science & Technology at the same institution. During the pursuit of this degree, the author had the opportunity of contributing to the research in the field of Systems Engineering. Njideka received her MS in Systems Engineering December 2007.