

7-2017

Improving Information Technology(IT) Risk Analysis with Resource Allocation

Bikrant Gautam

St. Cloud State University, bgautam1@stcloudstate.edu

Follow this and additional works at: https://repository.stcloudstate.edu/msia_etds

Recommended Citation

Gautam, Bikrant, "Improving Information Technology(IT) Risk Analysis with Resource Allocation" (2017). *Culminating Projects in Information Assurance*. 33.
https://repository.stcloudstate.edu/msia_etds/33

This Starred Paper is brought to you for free and open access by the Department of Information Systems at theRepository at St. Cloud State. It has been accepted for inclusion in Culminating Projects in Information Assurance by an authorized administrator of theRepository at St. Cloud State. For more information, please contact rswexelbaum@stcloudstate.edu.

Improving Information Technology (IT) Risk Analysis with Resource Allocation

by

Bikrant Gautam

A Thesis

Submitted to the Graduate Faculty of

St. Cloud State University

in Partial Fulfillment of the Requirements

for the Degree of

Master of Science

in Information Assurance

June, 2017

Thesis Committee:
Susantha Herath, Chairperson
Lynn Collen
Alexander Polacco

Abstract

The security of the computer network has become a challenging task with ever increasing attacks against it. The problem that lies ahead for management and security administrators is to allocate the resources in accordance with the pitfalls in their network systems. While traditional risk analysis can provide them the rough idea of what needs to be protected, an efficient method is imperative to plan the resources to tackle the incoming attacks in a more dynamic way. This paper focuses on modeling a traditional IT risk case into a mixed strategy game and help the security professionals to calculate and place their resources against the intruders with the help of game theory. The added advantage it provides over the traditional risk analysis is that it helps the administrator to re-allocate resources for the required level of safety for their assets. This paper paves the way for converting an existing IT risk analysis into a more robust logic based analysis which could prove beneficial in planning real world IT risk analysis and resource allocation.

Table of Contents

	Page
List of Tables	6
List of Figures	7
Chapter	
I. Introduction	9
Introduction	9
Problem Statement	10
Nature and Significance of the Problem	11
Objective of the Study	11
Study Questions/Hypotheses	12
Limitation of the Study	12
Summary	13
II. Background and Review of Literature	14
Introduction	14
Background Related to the Problem	14
Literature Related to the Methodology	16
Summary	27
III. Methodology	28
Introduction	28
Design of the Study	28
Data Collection Model	28

Chapter	Page
Tools and Techniques	28
Summary	36
IV. Implementation	37
Existing System	37
Proposed System	37
System Requirements	38
Software Environment	39
Summary	56
V. Analysis and Result	57
Home Page	57
Results	59
Summary	69
VI. System Design	70
Data Flow	70
Use Case and Sequence Diagram	72
Summary	73
VII. System Testing	73
Types of Testing	74
Test Objectives	77
Features Tested	77
Summary	78

Chapter	Page
VIII. Conclusion	79
References	80
Appendix	83

List of Tables

Table	Page
2.1 Attack Defend Game Example	25
2.2 Derivation of Mathematical Relation	26
3.1 Payoff Matrix for Attack Defend Model	31
3.2 Calculation of IT Risk Using Traditional Method	34
3.3 Combined Result of Traditional and Game Theory Approach	35

List of Figures

Figure	Page
2.1 OWASP Severity Score	19
4.1 Fusion Chart Gauge	52
4.2 Web Server Architecture	53
4.3 Browser Components	55
5.1 Application Home Page	58
5.2 Drop-Down Select Menu	58
5.3 Traditional Risk Score Input UI	59
5.4 Traditional Risk Score Output UI	60
5.5 Game Theory Text	61
5.6 Game Theory Input UI	62
5.7 Game Theory Risk Matrix	65
5.8 Game Theory Probability Distribution	63
5.9 Probability Radar Distribution	66
5.10 Resource Allocation Suggestion	64
5.11 Mixed Strategy Text	65
5.12 Mixed Strategy Risk Matrix	66
5.13 Mixed Strategy Probability Distribution	66
5.14 Mixed Strategy Radar Distribution	67
5.15 Equilibrium Probability Distribution	67
5.16 Traditional Risk Scoring of Devices	68

Figure	Page
5.17 Traditional Risk Scoring in Text	68
5.18 Conclusion	68
6.1 Application Data Flow Diagram	70
6.2 Application Sequence Diagram	72

Chapter I: Introduction

Introduction

The quantitative methodologies of determining a risk factor associated with the business are widely popular. While this model might hold true for varied systems, the complexity of business governed by information technology requires a very sophisticated risk analysis methodology for the defense of its business perimeter.

While a bank or a retail chain can have a known number of vulnerabilities, which can be monitored with the help of existing business risk analysis, the same could prove fatal while being used in IT security. New vulnerabilities are discovered almost every day. The number of Security Vulnerabilities for the month of August itself was over 100, with numerous vulnerabilities with CVSS scores greater than 7 (Security Vulnerabilities , 2016).

As the vulnerabilities are being discovered, rapid developments are made to address those problems with patches and updates coming from the vendors typically within a few days. A paradox arises when the existing vulnerability is automatically addressed with the software update, and a new vulnerability is introduced within a couple of hours, while the measures adopted by the company are still being employed for the nonexistent vulnerability.

The risk analysis for such cases need to be dynamic, with calculations being made in a quick interval of time with dynamic resource allocation for the incoming vulnerabilities. This paper discusses the possibility of employing a reasoning based

technique to address the risk management, in addition to the statistical or quantitative method.

When calculating IT risk where multiple and changing vulnerabilities exist, simple equations (Whiteman & Mattord, 2010) of multiplying likelihood values with the value of the asset, which mostly is based on conditional probabilities might not prove effective (Cox, 2009).

For years, security researchers have looked for alternative measures to address the problem faced in the field of security. Game theory is one of those fields, which bears maximum potential, and have been proposed as a risk analysis model against counterterrorism (Jesus Rios, 2012).

The calculation of the risk score provides a very basic set of information regarding the setting up of defense mechanisms for the risks. After evaluating the risk score using game theory technique, it will be easier for the defender to plan the defense and allocate resources considering the adversaries.

Problem Statement

The existing risk scoring mechanisms are only able to provide the rough idea of how vulnerable the system is. We still do not have a mechanism which provides the evaluation of available resources and helps the security professional to design the best possible defensive plan with the existing system.

The commonly used risk scoring mechanism simplifies the complexity of the problem with a borrowed risk determination equation from the traditional business model. However, considering the development of IT structure, the plain quantitative

method does not provide any additional information on how to place/plan the available resources.

Nature and Significance of the Problem

The current security measures only focus on the vulnerability of the system and suggestions regarding what could be added on top of available resources. Once security professionals find these risk scores, they plan the resource allocation without any reliable computation. This only makes the job half complete, as the resource allocation might still be inefficient making the whole IT infrastructure vulnerable.

The lack of smart measures of resource allocation persists the threats to the organization, and in certain instances, may worsen the scenario with the wrong resource divided between the risky systems.

Objective of the Study

The objective of this study as guided by the problem statement is to improve the IT risk analysis process with resource allocation.

This study will take consideration of various constraints pertaining to the vulnerable system, employ some advanced mathematical models and provide the rough idea of what amount of available resources can be applied for the best outcome in the system.

The goal for this discussion is to employ a competitive mixed strategy game theory in risk analysis for IT risk management and if possible, compare the results with the existing quantitative business risk analysis model. A similar work was done

by Louis Anthony Cox, Jr (2009), where he compared general risk analysis with game theory.

The objective of this discussion is to set an example of how implementation of game theory could substantially help the security professionals to make decisions over the resource allocation for the ever-changing vulnerabilities pertaining to IT field.

Study Questions/Hypotheses

The study question revolves around the methods that could be applied on the existing risk scoring method to calculate the resources allocated with the implementation of advanced and logical calculations.

The next step to improve the existing method is to adopt a well-researched and well-corroborated technologies, which has proven effective in analogous scenarios. The main study will try to show the implementation feasibility and effectiveness of resource allocation calculated using game theory techniques along with the existing risk scoring system.

Limitation of the Study

This study does not attempt to change the existing risk analysis method, but only suggests the newer approach towards achieving a better control over the Information Technology. The study is only a proposal towards adapting new technologies and could pose serious consequences if taken into a real time production environment.

Summary

This chapter provides the foundation of the objective for this study. The game theory technique has been a topic of interest in recent years. With the brief introduction of the research problem and the benefits provided from the game theory, we move into the next chapter where we will explore more on the literature review, mathematical derivation, and the use case scenarios of Information Technology risk calculation with game theory.

Chapter II: Background and Review of Literature

Introduction

The prerequisite of the game theory requires at least two intelligent parties capable of making an intelligent decision based on the scenario of the task, which would be favorable for each of them. Business dependent on IT employs security managers or security administrators who are responsible for the allocation of the resources against fending off the possible vulnerabilities that could exploit a system.

While the security experts have to defend all the vulnerable parameters with the limited resources available to them, the intruder only has to successfully exploit a single vulnerability to cause substantial damage. This can be analogous to a model of the game, where security admin and the intruder compete against each other where both try to optimize their move for their best benefit; the security administrator will focus on maximizing the mitigation against the probable vulnerability trying to be exploited, whereas the intruder will try to maximize the probability of a successful attack.

Background Related to the Problem

The cyber threat to the organizations (FBI, 2016) from the late 2000s have left researchers and security professionals wondering over the mechanism for the defenses (Strassmann, 2009). While the defense mechanism is well researched, a field left out is the analysis and consideration of the attacking model. An IT company might have a million-dollar worth of the latest firewalls to prevent any digital threat, but if they just employ a simple lock to close their main gate, any intruder with proper

information can trespass into their facility and transfer crucial information, physically present at the perimeter. The same analogy can be applied to their telephony or internet system.

To prevent any compromise against the Confidentiality, Availability and Integrity of the CIA triad, the weakest link should be strengthened along with other parameters. In the field of IT, the weakest link is a dynamic entity, while a system is patched and updated, another might fall into the hand of hackers.

The traditional, IT risk analysis methods only provide a numeric score based on the assumption of how vulnerable a system could be to the company. This vulnerability scoring does not account for the strength and the intelligence of the intruder party. The higher numeric score leads to the assumption that the system with a higher number needs more defenses, leaving the low score entities more exposed to the attackers, as the resources available for the defense is always finite in number. The hackers break into the organization's system to retrieve the information, from the least defended path. Once into the system, they can eavesdrop or escalate privileges to gain access to the crucial information.

The traditional method does not count for the correlation among the components and fails to assign resources accordingly. The traditional method is a probabilistic risk assessment technique that ignores the dynamic features like adaptation and planning.

The limitation of traditional risk scoring methods can be improved by adopting the game theory technique. While calculating the risk associated with a system, it

also considers the moves of the opponent to estimate the likelihood of any attacks. This can be improved by creating a system which takes live parameters and provides the estimate of the strength of the attacking parties.

Literature Related to the Methodology

Different risk rating methodologies.

CVSS. Common Vulnerability Scoring System (CVSS) is the collaborative initiative initiated by the U.S. Department of Homeland Security (DHS) involving various IT Security giants such as Cisco Systems, Symantec, ISS, Qualys, Microsoft, CERT/CC and eBay (OWASP, 2017). This group has collaborated in multiple projects and CVSS was one of the outputs.

The advantages of CVSS:

- Provide accurate and normalized severity rating for the vulnerabilities in the system. Helps to alert the customer to the appropriate action required.
- Help security researchers to find several threats and exploits in their systems. The CVSS ranking system produces reliable risk scoring to ensure that the exploits will be taken seriously per their ratings.
- CVSS has been recommended by the working group for use by U.S. Government departments.

The limitations of CVSS:

- CVSS fails to reduce the attack surface area like design flaw, or help compute risks within any arbitrary piece of code. It is a plain scoring system and cannot be taken as a risk modeling methodology.

- CVSS is more complex than other risk scoring systems, as it tries to calculate the risk of announced vulnerabilities as present in the running software systems and other environment variables.
- The CVSS risk ranking is a complex process. The security researchers need to prepare a spreadsheet to calculate the risk components, as soon as a new vulnerability like a worm or Trojan has been released targeting a small number of attack vectors.
- The overhead of calculating the CVSS risk ranking is quite high if applied to a thorough code review, which may have 250 or more threats to rank.

OCTAVE. OCTAVE is a heavyweight risk methodology approach which involves advanced computation. OCTAVE originated from the Carnegie Mellon University's Software Engineering Institute (SEI) in collaboration with CERT (CERT, 2017). OCTAVE focuses on organizational risk, not technical risk. OCTAVE comes in two versions: Full OCTAVE, for large organizations, and OCTAVE-S for small organizations, both of which have specific catalogs of practices, profiles, and worksheets to document the modeling outcomes.

Advantages of OCTAVE:

- Implementing an organizational culture of risk management and controls becomes necessary.
- Documenting and measuring business risk becomes timely.
- Documenting and measuring the overall IT security risk, particularly as it relates to the corporate IT risk management, becomes necessary.

- When documenting risks surrounding complete systems becomes necessary.
- To accommodate a fundamental reorganization, such as when an organization does not have a working risk methodology in place and requires a robust risk management framework to be put in place.

The limitations of OCTAVE:

- OCTAVE is incompatible with available standards such as AS/NZS 4360, as it assumes a threat will always occur and this is inappropriate for many organizations. OCTAVE-S makes the inclusion of this probability optional, but this is not part of the more comprehensive OCTAVE standard.
- Consisting of 18 volumes, OCTAVE is large and complex, with many worksheets and practices to implement.
- It does not provide a list of “out of the box” practices for assessing and mitigating web application security risks.

Because of these reasons, the Open Web Application Security Project (OWASP) does not anticipate that OCTAVE will be used at large by application designers or developers, because it fails to take threat risk modeling into consideration, which is useful during all stages of development, by all participants, to reduce the overall risk of an application becoming vulnerable to attack.

OWASP Risk Rating. Open Web Application Security Project (OWASP) is an online community which creates freely-available articles, methodologies, documentation, tools, and technologies in the field of web application security. The

OWASP risk rating is calculated in 6 different steps (OWASP, 2017). Those steps are:

- Identify the risk. The security personnel collect information about the threat agent involved, exploit being used and the vulnerabilities involved.
- Factors for Estimating Likelihood. Once the security researcher has identified the risks, the measure of how likely the vulnerability is going to occur is calculated.
- Factors for Estimating Impact. The business impact and the technical impact of the attack affect the organization. Hence all the details about the technical and business risk should be collected to decide about the risk.
- Determining Severity of the Risk. In this step, the overall severity of the risk is calculated. This is done by estimating whether the likelihood is low, medium, or high. The scale for it is split into the following parts.

Likelihood and Impact Levels	
0 to <3	LOW
3 to <6	MEDIUM
6 to 9	HIGH

Figure 2.1: OWASP Severity Score

- Deciding what to fix. After the risks are classified, the risks must be prioritized. As a rule, the most severe risk should be fixed first and other risks come accordingly.

- Customizing the Risk Rating Model. A customizable risk rating system helps a business as it adapts per the environment. A well-designed model is more likely to provide the exact result whenever the risks occur.

Game theory. Security in the computer network has been an issue in past decades. This field itself is gathering interest from many different researchers. However, the problem has been addressed to a very limited extent. With advanced mathematical modeling, researchers are trying to come up with a solution in IT related to the field using game theories. This section will discuss a few of the game theory solutions proposed by the researchers to improve network security.

Game Theory. “A course in game theory” (Osborne & Rubinstein, 1994) defines a game as:

A game is a description of strategic interaction that includes the constraints on the actions that the players can take and the players’ interests, but does not specify the actions that the players do take. A solution is a systematic description of the outcomes that may emerge in a family of games. Game theory suggests reasonable solutions for classes of games and examines their properties.

Game theory is described as a multi-party interaction, which includes decision-making scenarios for the maximum gain of the involved parties. While doing so, the player chooses the course path, which will result in the best outcome for self, while expecting the concurrent loss to the adversary.

A player is the basic entity of the game that makes rational decision and then executes the actions accordingly. While a game is a strategic interaction that involves various constraints. A solution concept is a systematic description of how the decisions are taken to play the game resulting in the best possible outcome. The consequence function associates a consequence with each action the player takes. A preference relation is a complete relation on the set of consequences which model the preference of each player. A strategy for a player is the complete set of plans of actions in all possible situations through the game. If the strategy employed takes a unique action in a situation, then it is called a pure strategy. If the plan specifies a probability distribution for all possible actions in a situation, then that strategy is referred as a mixed strategy.

A Nash equilibrium is a solution concept that describes a steady state condition of the game; no player would prefer to change his strategy as that would lower his payoffs given that all other players are adhering to the prescribed strategy. This solution concept only specifies the steady state but does not specify how that steady state is reached in the game. The Nash equilibrium is the most famous equilibrium, even though there are many other solution concepts used occasionally. This information will be used to define games that have relevant features for representing network security problems (Roy et al., 2010).

A *player* is a decision-making entity that drives the output of the game. Security personnel, hacker(s) or even machine or software employed, that make strategic moves in the game can be a player.

An *action* is a move made by the player in the game. *Payoff* is the consequence for the action carried out by a player in the game. *Strategy* is the method of the gameplay that the player chooses to excel the adversary.

A *Payoff Matrix* is a matrix of size $M * N$ and includes the possible outcome each player with each player having M and N possible moves respectively. We will make use of this matrix for the determination of the optimal strategy for the risk analysis and mitigation.

Expected Utility is the resultant of the strategy chosen by another player. The utility is the function of opponent's payoff and the probability of selecting an option.

Mixed strategy is when players chose their options in random and when no pure-strategy equilibria (Nash, 1999) exist.

Payoff is a positive or negative reward to a player for a given action within a game.

Strategy is a plain of action within the game that a player can use.

Perfect Information Game is a game in which each player is aware of the moves of the adversary.

Imperfect Information Game is a game where a player does not know the move of another player.

Complete Information Game is a game in which every player knows the strategies and payoffs but not necessarily the actions.

Bayesian Game is a game in which the information about the strategies and payoff for other players is incomplete and a player assigns a type to the other players.

Static/Strategies Game is a one-time game in which each player chooses his plan of action and all player's decision are made at the same time, simultaneously.

Dynamic/Extensive Game is a game with more than one stages in which the players can make their moves.

Stochastic Game is a game that involves probabilistic transitions through several stages of the system. The game consists of several states of the system.

Derivation of the Attack Defend Model. We will attempt to model a simple scenario of game theory with a basic mathematical explanation for the understanding of how it works. The scenario is similar to other studies (Cox, 2009; Jesus Rios, 2012) which have attempted to depict the simple analysis of game theory and risk management for counterterrorism modeling.

1. A security administrator **A**, for the XYZ Company, allocates different resources for Information Technology infrastructure after the vulnerability assessment.
2. Intruder **D**—who has the knowledge of the vulnerabilities of the target IT system—allocates his resources for the attack, considering the defender's strategy of resource placement.
3. Each strategy for each player has a consequence as the loss of property for their investment. These values are modeled as a random variable with some

probability distribution which is dependent on the allocations made by the players (Cox, 2009).

Modeling the game as described above relieves us from the discussion of complex game theory concepts as Perfect game, Dynamic/Stochastic, Bayesian Nash, etc. (Cox, 2009). L. A. Cox (2009) on his paper has suggested that:

Relatively simple optimization can be used to solve for the attacker's best response to any allocation of defensive resources and solve for defender's best allocation of resources considering the attacker's best response. Hence it is possible to solve different attacker-defender game using simple optimization method, without involving complex game theoretic concepts and terminologies.

Assuming that those vulnerabilities have been discovered and the management or the security team has identified the targets that could pose a risk as target **X** and target **Y**.

This calculation makes use of a mixed strategy where the players choose randomly among their available options. Even though the players have made the selection of sets for their moves considering each other's strategies, the way they select each strategy is a pure random function; there is an equal probability for an attacker to attack any of the target X and Y with one of the strategies from his resource set.

Let us see a simple two-stage payoff matrix for the attacker and the defender. After the values are set by the player, the game is played more like as a *minimax*

algorithm (Stanford University, 2016), where each player makes a move to minimize his adversary's strategy.

A simple payoff matrix for the attacker-defender model is as follows:

Table 2.1: Attack Defend Game Example

	Attack X	Attack Y
Defender's selection (row)	Attacker's selection (column)	
	20, -10	80, -10
	-40, 20	-5, 20

From Table 2.1, for target **X**, the defender has employed a resource that benefits him with a value of 20. For the same asset, the attacker will bear a loss of 10 consequently if that game is played. Similarly, for another case, the defender bears a loss of 40 while the attacker gains 20.

For target **Y**, the defender has employed a defensive resource, which benefits him 80, while causing the attacker the loss of 10. Similarly, for another case the attacker has allocated his resource which earns him 20 and causes the defender the loss of 5.

Let us take a generic payoff matrix (Table 2) and derive the probability of a successful event:

Table 2.2: Derivation of Mathematical Relation

	Intruder selects column	
Defender selects row	a_1 , a_2	b_1 , b_2
	c_1 , c_2	d_1 , d_2

Let us define the expected utility for the attacker:

The expected utility for attacker **A** attacking target X is U_x

The expected utility for attacker **A** attacking target Y is U_y

Both the utility is the function of the probability distribution (P_D) that the player D will play with a *mixed strategy* (Spaniel, 2016).

Mathematically we can represent U_x and U_y as;

$$U_x = F(P_D) \quad (i)$$

$$U_y = F(P_D) \quad (ii)$$

For player D willing to mix, the probability distribution (P_D) exists such that;

$$U_x = U_y \quad (iii)$$

That is, attacker A's utility of attacking target X is equal to the attacker's utility of attacking target Y (Spaniel, 2016).

Calculating the utility for the attacker when the defender is willing to play and the attacker chooses to attack:

$$U_x = a_2(P_D) + c_2(1 - P_D) \quad (iv)$$

$$U_y = b_2(P_D) + d_2(1 - P_D) \quad (v)$$

From equation (i), (ii), (iii), (iv) and (v) we can write;

$$a_2(P_D) + c_2(1 - P_D) = b_2(P_D) + d_2(1 - P_D)$$

Solving for P_D , we get:

$$P_D = (d_2 - c_2) / [(d_2 - c_2) + (a_2 - b_2)]$$

We have calculated the probability that the defender will defend his perimeter, from the payoff matrix, similarly calculating the probability for the attacker (P_A) we get,

$$P_A = (d_1 - b_1) / [(d_1 - b_1) + (a_1 - c_1)]$$

Summary

This chapter established the relation and understanding of mixed strategy for two competing entities. In the next chapter, we will use the relation from this chapter into a theoretical implementation on different scenarios that might occur in an IT organization.

Chapter III: Methodology

Introduction

This chapter includes the implementation of game theory techniques for evaluating the risk score associated with different entities of an organization. We will also see the comparative study of game theory technique with the traditional risk scoring system.

Design of the Study

The research methodology for this study will primarily be the quantitative approach, with mathematical calculations and comparisons between risk score calculating relations. The quantitative approach suits this study better, as we rely on the mathematical and numeric figures to compare and contrast the result.

Data Collection Model

In this section, the mathematical relation derived from the earlier chapter is implemented on a few case scenarios, along with the comparison of the traditional risk scoring model. The objective of this section is to provide the numeric data from the empirical method for the implementation of a Computer Program, which helps to calculate the risk score based on game theory and traditional techniques.

Tools and Techniques

Game theory model. Suppose the security administrator was notified that two vulnerabilities exist on his network; a probable virus outbreak on the Mail Server (case **A**) and denial of service at their Web Server (case **B**).

The intruder has the information of these vulnerabilities, simply discovered by a post on a hacker's forum on Reddit (/r/Hacking, 2016), where someone had posted the result of the penetration testing of that company. This scenario will take the form of a game played between the defender and the attacker where both of them allocate their resources for maximizing their objectives.

Let's try to convert a traditional risk analysis case (Whiteman & Mattord, 2010) into a game payoff matrix using an example scenario:

- Information asset **A** has a value score of 50 and has one vulnerability.
Vulnerability 1 has a likelihood of 1.0 with no current controls. You estimate the assumptions and data are 90% accurate.
- Information asset **B** has a value score of 100 and has one vulnerability.
Vulnerability 2 has a likelihood of 0.5 with current controls addressing 50% of its risk.

Scenario 1 informs us that the value of asset A is 50, and no defensive measures are employed. This means asset A will incur a loss of 50 when under attack.

Let us assume that the attacker employs 10 units of his resources to infect the target system. The total benefit for the hacker, in this case, will be 40.

For the payoff matrix, an administrator has to play a game between the selection of resources for, the best case and the worst case.

Let us do additional work for the security admin and assume that we have applied controls and it addresses 50% of the risk. Likewise, the security administrator becomes successful in preventing the asset, let's assume the benefit for this move

will earn 25 value for asset A. When the security administrator employs some defensive mechanism, the intruder will have some disadvantage, either he will get caught or he will have to allocate more resources to compromise the system. Let us assume that the intruder will face a 35 unit loss when attempting to break into the protected system.

The value of asset B is 100 with a vulnerability likelihood of .5 with a control addressing 50%. The benefit for asset B will be only 50. Similarly, let us assume the intruder will face a 35 unit loss when attempting to break into the web service application.

Let us assume, the security administrator decides to re-organize all of the defense mechanisms to prevent an attack against B with no controls. In the case of a successful denial of service attack, the loss incurred by B will be 100. Let's assume that the hacker employed 10 units of resources to compromise system B, hence his maximized benefit is 90.

Feeding this data into the pay-off matrix we get:

Table 3.1: Payoff Matrix for Attack Defend Model

		Attack A	Attack B
Defender's selection		Attacker's selection (column)	
	Defend A	25, -35	-100, 90
	Defend B	-50, 40	50, -35

A is exposed with zero controls in place hence the defender will incur the loss of 50, whereas the attacker for the same case gains 40, deducting his investment. When defensive resources are increased at **A** by 50%, the benefit is maximized to 25, but the attacker will also have to increase his resources to compromise the target, assume the attacker bears a loss of 35 units of resources while compromising this system.

B has vulnerability and some control. **B** will be able to maximize the half of its asset, while the hacker will have to risk his resources to compromise this system, let's assume that value to be 35. When defensive resources at **B** are completely removed, the loss incurred by B will be 100, the benefit for the attacker will be 90 with 10 as his investment for compromising the system.

The value of P_A and P_D calculated from the payoff matrix are as follows:

$$P_D = \frac{3}{8}$$

$$P_A = \frac{2}{3}$$

This explains that the defender will protect asset A with a probability of $\frac{5}{8}$ and protect asset B with a probability of $\frac{3}{8}$.

The probability of defense explains that the attacker will be indifferent to the attack target A or B, given the probability of defense. If the defender wishes to increase the defense at A, he should increase the probability of protecting it by more than the calculated value, which will refrain the attacker from making the move towards A. The expected utility when the attacker attacks A or B is 11.875.

Similarly, if he wants to deter the intruder, the defender should increase his defense at B making the defensive probability higher at B.

Also, the attacker will make an attack at A with the probability of $\frac{2}{3}$ and attack B with the probability of $\frac{1}{3}$. The expected utility when the defender defends **A** or **B** with indifference the attack probability is -16.667.

The probability of the attack, also called as the threat can be used to calculate the optimum strategy. If the player deviates from this strategy, he only loses while the adversary can gain. In the above case, the defender can expect the total loss of 16.667, however, if he changes his strategy with his existing resource allocation, his loss will increase.

Thus, game theory provides the security planners with an idea of resource allocation with consideration of threat probability.

A Traditional Model. The following equation represents one of the widely taught risk scoring formulas used for the risk value determination.

$$\text{Risk} = (L * V) - C + U \quad (\text{Whiteman \& Mattord, 2010})$$

Where,

L is likelihood of an occurrence of a vulnerability

V is value of the information asset

C is percentage of risk mitigated by current controls

U is uncertainty of current knowledge of the vulnerability

The likelihood as defined by the NIST SP 800-30 is a rating between 0.1 and 1.0, which are mostly assumed before calculating the risk score. Apart from certain events which can bear probability 1, there is no other method to find the exact likelihood of an event for a scenario.

The V and C are calculated values which will approximate towards the actual risk scoring, however, U is another assumption made by the security manager based solely on judgment and experience (Whiteman & Mattord, 2010).

Applying the traditional risk scoring formula for the scenario discussed above;

Risk rating for **A** is 55.

Risk rating for **B** is 35.

This tells us that A is more vulnerable than B, however, it does not provide any suggestion or medium to calculate the resource allocation for the defense. A cursory inspection leads us to believe A is more vulnerable, which might encourage the security administrator to allocate defensive resources for B to A, however, the asset

value of B is more than A, and leaving it exposed will cause more damage to the institution. Allocating more resources to A than B considering the risk score might prove disastrous as it would be the waste of resources, and adding to that, an exposed valuable asset B.

Mixed Model. Tabulating the earlier findings for Asset **A** and Asset **B**.

Table 3.2: Calculation of IT Risk Using Traditional Method

	Value	Likelihood	Control	Uncertainty	Risk Value
Asset A	50	1.0	0	10%	55
Asset B	100	1.0	50%	0	25

Table 3.3: Combined Result of Traditional and Game Theory Approach

	Traditional	Game Theory (Defense, Attack) probability
Asset A (value 50)	55	$(\frac{5}{8}, \frac{2}{3})$
Asset B (value 100)	25	$(\frac{3}{8}, \frac{1}{3})$

The traditional score from the Table 3.3 only provides the risk rating of each asset. Which, in many cases could be insufficient to make a logical decision to place the defensive resources.

Now when taking account of defense and attack probability on individual assets, the security administrator can have a rough estimate of payoff for every move he makes while allocating resources.

The above table has simplified the real world problem, however, for real case scenarios this matrix would contain numerous assets with different risk ratings and attack/defense probabilities. Considering multiple vulnerabilities and limited resources available, the network admin first could have the rough estimation of the critical systems under his supervision with the help of a traditional risk analysis method. The traditional risk analysis can be employed to create a cut-off value for the

risks. Once identified, he could then pick the most critical systems and apply game-theory analysis to place his resources accordingly.

In above table risk score of asset A is higher compared to asset B. However, these scores are not enough to allocate defensive resources. It seems that asset A has a greater risk score so the security administrator might be tempted to allocate more resources to it. Since it does not take into consideration asset B, which even though it has a relatively low-risk value can cause more damage to the organization if it was successfully exploited.

However, considering the game played between the security administrator and the intruder, now the security administrator has an idea of how many resources of the total allocation can be assigned to each asset. For defending asset B, he must increase his defense strategy to decrease the attack probability on that specific target. Combining the traditional approach of IT risk analysis with game theory will help to first;

- Identify the vulnerable resources.
- Help allocate resources to the specific assets.

Summary

This chapter discusses the mathematical modeling of the risk analysis cases involving traditional, game theory and mixed approach.

Chapter IV: Implementation

Existing System

Information Technology risk analysis turns out to be a high priority issued in IT security and assurance. The existing risk scoring method seems to lack the proper resource allocation methodologies, which in turn produces more redundancies and inefficiencies in the system.

In a practical scenario, the security professional would like to know how many resources should be allocated to a system after discovering the risk score associated with it. Traditionally, the highest scoring devices will get most of the resources, and the less scoring devices will get fewer of the resources. However, a proper logical computation of resource allocation would relieve the security professionals from randomly estimating the resources to the systems, thus providing a better defensive plan compared to the conventional approach.

Proposed System

In the proposed system, the users are provided the additional information regarding the allocation of resources for the same input parameters they provide for the existing system.

This will drastically reduce the complexity on the user and they will be able to get a better grasp of the current situation of their IT devices. All the calculations are done under the hood, leaving users with a very intuitive user interface with recommendations, statistics and data visualization over how resources can be planned according to the new system.

The proposed system will evaluate the existing risk score using the traditional method. It will also calculate the resource allocation using game theory and finally it will give a smart logical reporting over how defensive mechanism can be distributed over the available devices.

The result of this research is to help the security professionals to visualize the benefit of the solution provided by this paper. The proposed system will include the theoretical background of each method, and explanation of what different scores mean and a graphical representation of various data.

System Requirements

The hardware and software required for the comparison of the study are as follows:

Hardware requirements.

System : Pentium IV 2.4 GHz.

Hard Disk : 500 GB.

Monitor : Any (1).

Mouse : Any (1).

Keyboard : Any (1).

Ram : 2 GB.

Software requirements.

Operating system : Windows 7/Linux.

Coding Language : jQuery UI, jQuery, FusionChart.JS

IDE : Bracket

SERVER	: Apache Tomcat 7
HOST	: GitHub
Browser	: Chrome/IE/Firefox

Software Environment

This chapter explains in detail what software environment was required and discusses the implementation of the code for the execution of the project. The implementation of this project has been realized as a web application. The web application runs in a remote web server which is accessible to anyone connected to the internet. The software includes latest technologies, which provide robust analysis mechanisms to view the data.

In addition to computation, the software also provides an easy overview and comparison and contrast between the operations suggested in this starred paper.

Web Technologies

Web technology was used to implement the project, as traditional software would require installing the application on each individual computer. Once the files were hosted in the GitHub cloud, the project was readily available to any devices: mobile, PC or tablet connected to the internet. Web technologies, though complex and hard to implement compared to the traditional standalone software, provides broader outreach and helped to improve the implementation.

The web technologies were used against the traditional software development method for the following key points:

- Cost effective development
- Accessible anywhere
- Easily customizable
- Accessible for a range of devices
- Improved interoperability
- Easier installation and maintenance
- Adaptable to increased workload
- Increased security
- Flexible core technologies
- Easier to install and maintain
- More useful to the users.

The modern web technologies are responsive in design, meaning regardless of the platform used they will adapt to the user's system for easy usage. Web technologies offer a wide variety of different languages running in the backend, making it easy for users to use the application regardless of what platform it was built on.

HTML5. HTML5 is a markup language used for structuring and presenting the world-wide web. HTML5 is the fifth version of the HTML standard. HTML5 was published on 2014 by the World Wide Web Consortium. HTML5 represents the attributes and elements of the modern website. HTML5 saw some enhancement on the HTML scripts with the deprecation of different elements from HTML4. HTML5 provides a more dynamic way of designing web pages with seamless integration with

Cascading Style Sheet and Java Script. HTML5 supports multiple new Application

Program Interfaces (API) such as:

- Canvas
- Timed Media Playback
- Offline
- Editable content
- Drag-and-drop
- History
- MIME type and protocol handler registration
- Microdata
- Web Messaging
- Web Storage

JavaScript. JavaScript is a dynamic computer programming language. It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side scripts to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities. JavaScript was first known as LiveScript, but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name LiveScript. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers.

The ECMA-262 Specification defined a standard version of the core JavaScript language.

- JavaScript is a lightweight, interpreted programming language.
- Designed for creating network-centric applications.
- Complementary to and integrated with Java.
- Complementary to and integrated with HTML.
- Open and cross-platform

The logic for risk calculation in this paper is completely implemented using JavaScript and its library. While the rendering of the charts and the User Interface includes external libraries, the author has designed his own implementation in code for calculating all the risk. Following is a script example from the custom JavaScript for calculating the risk score using the game theory methodology.

```
if (value == 2) {
    $("#traditionalCheck").slideUp(300, "swing");
    $("#mixedCheck").slideUp(300, "swing");
    $("#gameTheoryCheck").slideDown(300, "swing");
    $("#gameTheoryTheory").hide();
    $("#gameTheoryTheory").show("bounce", {
        times: 2
    }, "slow");
    $("#gameForm").submit(function (event) {
        var isvalidate = $("#gameForm").valid();
        if (isvalidate) {
```

```
console.log("works so far");
```

```
var name1 = $("#deviceName1").val();
```

```
var asset1 = $("#asset1").val();
```

```
var risk1 = $("#risk1").val();
```

```
var uncertainty1 = $("#uncertainty1").val();
```

```
var control1 = $("#control1").val();
```

```
var Afav = calculateTraditional(+asset1, +risk1 / 100, +control1, +uncertainty1);
```

```
var AttackUnfav = -10 - Afav;
```

```
var Aunfav = 0 - (+asset1);
```

```
var Attacfav = (+asset1) - 10;
```

```
var name2 = $("#deviceName2").val();
```

```
var asset2 = $("#asset2").val();
```

```
var risk2 = $("#risk2").val();
```

```
var uncertainty2 = $("#uncertainty2").val();
```

```
var control2 = $("#control2").val();
```

```
var Bfav = calculateTraditional(+asset2, +risk2 / 100, +control2, +uncertainty2);
```

```
var BttackUnfav = -10 - Bfav;
```

```
var Bunfav = 0 - (+asset2);
```

```
var Bttacfav = (+asset2) - 10;
```

```

var PD = (BttackUnfav - Attacfav) / ((BttackUnfav - Attacfav) + (AttackUnfav - Bttacfav));

PD = PD.toFixed(2);

var PDnot = 1 - PD;

var PA = (Bfav - Bunfav) / ((Bfav - Bunfav) + (Afav - Aunfav));

PA = PA.toFixed(2);

var PAnot = 1 - PA;

location1 = "#attackProbability";

location2 = "#defendProbability";

caption1 = "Attack probabailiy Distribution";

caption2 = "Defend probabailty Distribution";

locationDiv = "#radarChart";

var riskScore = calculateTraditional(+asset1, +risk1 / 100, +control1, +uncertanity1);

$("#gameText").html("<h3>The risk matrix for given sets of devices is</h3>" +
"<table align=\"center\" style=\"border: 1px solid black; text-align:center\"><tr><th></th><th></th><th></th></tr><tr><th class=\"tableRed\">" + name1 + "</th><th class=\"tableRed\">" + name2 + "</th></tr>" +
"<tr><td class=\"tableGreen\">Defend " + name1 + "</td><td class=\"tableCell\">" +
Afav.toFixed(2) + " " + AttackUnfav.toFixed(2) + "</td>" +
"<td class=\"tableCell\">" + Bunfav.toFixed(2) + " " + Bttacfav.toFixed(2) + "</td></tr>" +
"<tr><td class=\"tableGreen\">Defend " + name2 + "</td><td class=\"tableCell\">" +
Aunfav.toFixed(2) + " " + Attacfav.toFixed(2) + "</td>" +
"<td class=\"tableCell\">" + Bfav.toFixed(2) + " " + BttackUnfav.toFixed(2) +
"</td></tr></table><br><br>" +

```

```
"<p class=\"gameResultFinal\">Defense Probability of device: " + name1 + " is " + PD + "<br>
Defense Probability of device: " + name2 + " is " + (1 - PD) + "</p><br>" +
"<p class=\"gameResultFinal\">Attack Probability on device: " + name1 + " is " + PA + "<br>
Attack Probability on device: " + name2 + " is " + (1 - PA) + "</p><br>";
```

```
$("#gameChartContainer").slideDown(300, "swing");
drawGame(PA, PAnot, name1, name2, location1, caption1);
drawGame(PD, PDnot, name1, name2, location2, caption2);
drawRadar(PA, PAnot, PD, PDnot, name1, name2, locationDiv);
$("#gameFinalText").show();
$("#gameFinalText").html("<p style=\"line-height: 30px;\">To maintain the equilibrium between
attack and defense the device: <span class=\"myButton\">" + name1 +
"</span> should be allocated <span class=\"myButton\">" + (PA*100).toFixed(2) + "%</span>
of the available resources and device: <span class=\"myButton\">" + name2 +
"</span> should be allocated <span class=\"myButton\">" + (PAnot*100).toFixed(2) +
"%</span> of the available resources.</p>");
```

jQuery. jQuery is a cross-platform JavaScript library designed to simplify the client-side scripting of HTML (jQuery, January 23, 2017). jQuery is the most popular JavaScript library in use today, with installations on 65% of the top 10 million highest-trafficked sites on the Web (Libscore, 2017). jQuery is free, open-source software licensed under the MIT License (JS Foundation, 2016).

jQuery is designed to make it easier to navigate a document, select DOM elements, create animations, handle events, and develop Ajax applications. jQuery

also provides capabilities for developers to create plug-ins on top of the JavaScript library. This enables developers to create abstractions for low-level interaction and animation, advanced effects and high-level, theme-able widgets. The modular approach to the jQuery library allows the creation of powerful dynamic web pages and Web applications (jQuery, January 15, 2017).

For this project, jQuery was the best option as implementation of the algorithm could be done in an easy manner. For the demo, we didn't have to make a connection to any database engines and jQuery proved to be very robust and a faster programming language, as it only required a web browser to run its code. jQuery is comparatively easy to use as compared with native JavaScript with a large set of libraries. Also, the community around jQuery is very large and provides instant support. For this project and confusion regarding the development was easily overcome with the documentation and tutorials provided by the jQuery team. jQuery must be included in the related HTML file, generally it is called at the top of the HTML element. But if there are complex JavaScript functions operating over the values, then it is advised to call the java scripts at the very bottom of the HTML page. jQuery can be pointed directly at the HTML without downloading or using the Content Distribution Network (CDN) address as follows:

```
<script  
src="https://ajax.googleapis.com/ajax/libs/jquery/1.8.3/jquery.min.js"></script>
```

Once included, now we can write the java scripts using the jQuery library as:

```
<script>
```

```

$(document).ready(function () {

    $("#traditionalCheck").hide();

    $("#gameTheoryCheck").hide();

    $("#mixedCheck").hide();

    $("#traditionalTheory").hide();

    $("#gameFinalText").hide();

    $("#mixedFinalText").hide();

    $("#traditionalText1").hide();

    $("#refreshTraditional").click(function () {

        $("#traditionalForm")[0].reset();

        $("#traditionalText").text("");

        $("#messageBox").text("");

        $("#gaugeContainer").hide();

    });

</script>

```

\$ sign in JavaScript signifies that it is a jQuery operation. The more standard usage and practice is to include all the JavaScript functions within `$(document).ready(){ . . . }`; as the JavaScript function will only start to operate after the page has loaded completely. There could be other instances where users might want to run the function before the page loads, and for that the jQuery function can be kept out of this element.

jQuery UI. jQuery UI provides abstractions for low-level interaction and animation, advanced effects and high-level, theme-able widgets, built on top of the

jQuery JavaScript Library, that you can use to build highly interactive web applications (jQueryUI, 2017).

jQuery bundles different widgets, visual effects and themes, which are implemented using jQuery JavaScript library, Cascading Style Sheets and HTML. jQuery is very popular and there are over one million websites that make use of the jQuery UI. Notable users include Pinterest, PayPal, IMDB, The Huffington Post, and Netflix (Libscore, 2017). Both jQuery and jQuery UI are free and open-source software distributed by the jQuery Foundation under the MIT License; jQuery UI was first published in September 2007.

The front end of this project is designed with jQuery UI using different libraries bundled along with jQueryUI.js. jQuery UI provides many features that a professional web application has, including the ease and simplicity of use. The hide/show along with different animation options help to create a very user-friendly and intuitive application which is easy to use and understand.

jQuery UI can be included as follows in the HTML using CDN address.

```
<script src="http://code.jquery.com/ui/1.9.2/jquery-ui.js"></script>
```

jQuery UI works along with the CSS that comes along with the jQuery UI. The CSS includes different styles of presenting the elements of the web page.

Fusion Chart JS. Fusion Charts, part of InfoSoft Global (P) Ltd, is a privately held software provider of data visualization products (JavaScript Charts, Maps, Widgets and Dashboards) with offices in Bangalore and Kolkata, India. The company's flagship product and namesake, Fusion Charts Suite XT, is used by over

80% of Fortune 500 companies. Notable fusion chart customers are Apple, Google, ZOHO, Cisco, Facebook, Intel, LinkedIn, Microsoft, Hewlett-Packard, IBM, EMC, Nokia, Tibco, as well as The Weather Channel, NASA, and the Federal Government of the United States (Mitra, 2017).

This project contains various functions of the fusion charts like gauge, radar chart, doughnut chart etc. The integration of Fusion Chart with JS is seamless and easy for developers. While the version used for this project is the trial version, all the functionalities were available making it a very robust platform for professional looking chart development.

Fusion chart also needs to be included in the HTML script for the functioning. Users can download the fusioncharts.js from Fusion Chart and use the available library in their function.

```
<script type="text/javascript" src="js/fusioncharts.js"></script>
<script src="js/fusioncharts-jquery-plugin.js"></script>
```

Example of Fusion Chart implementation:

```
function drawGauge(riskScore, deviceName, locationGauge) {
$(locationGauge).insertFusionCharts({
    type: 'angulargauge'
    , width: '400'
    , height: '250'
    , dataFormat: 'json'
    , dataSource: {
        "chart": {
```

```

    "caption": "Traditional Risk Score"
    , "subcaption": deviceName
    , "lowerLimit": "0"
    , "upperLimit": "100"
    , "lowerLimitDisplay": "Good"
    , "upperLimitDisplay": "Bad"
    , "showValue": "1"
    , "valueBelowPivot": "1"
    , "theme": "fint"
  }
  , "colorRange": {
    "color": [
      {
        "minValue": "0"
        , "maxValue": "50"
        , "code": "#6baa01"
      }
    ]
  }
  , {
    "minValue": "50"
    , "maxValue": "75"
    , "code": "#f8bd19"
  }
  , {

```

```

        "minValue": "75"
        , "maxValue": "100"
        , "code": "#e44a00"
    }
]

    }
    , "dials": {
        "dial": [{
            "value": riskScore
        }
    ]
    }
}
});
};

```

This function draws the gauge as follows from the given parameters. The invocation of the chart function is very easy. The user will have to specify the id of the div where the chart should be drawn, the type of chart being drawn and pass the set of data to this function.



Figure 4.1: Fusion Chart Gauge

Apache Tomcat. Apache Tomcat Server is an open source web server developed and supported by the Apache group. It consists of the servlet container which will be used by the java servlets and java server pages (JSP) technologies as a reference for implementation. These servlets, JSPs and their specifications are developed by Java Community Process under the collaboration of Sun Microsystems. Unlike the traditional application servers like WebLogic, Tomcat is a web server and supports applications that are built using any programming language and any IDE. Tomcat is used to run the web applications on the host and acts as a local server that is built on the port 8080. It is composed of a web container named Catalina and bin directory. It can initiate the response methods or objects like GET and POST after loading all the HTTP related requests. The completed project was first hosted on a locally run tomcat server and then later transferred into GIT hub pages for public access.

The following figure shows the basic functionality of a web server and how it works along with the browser to present the data and services.

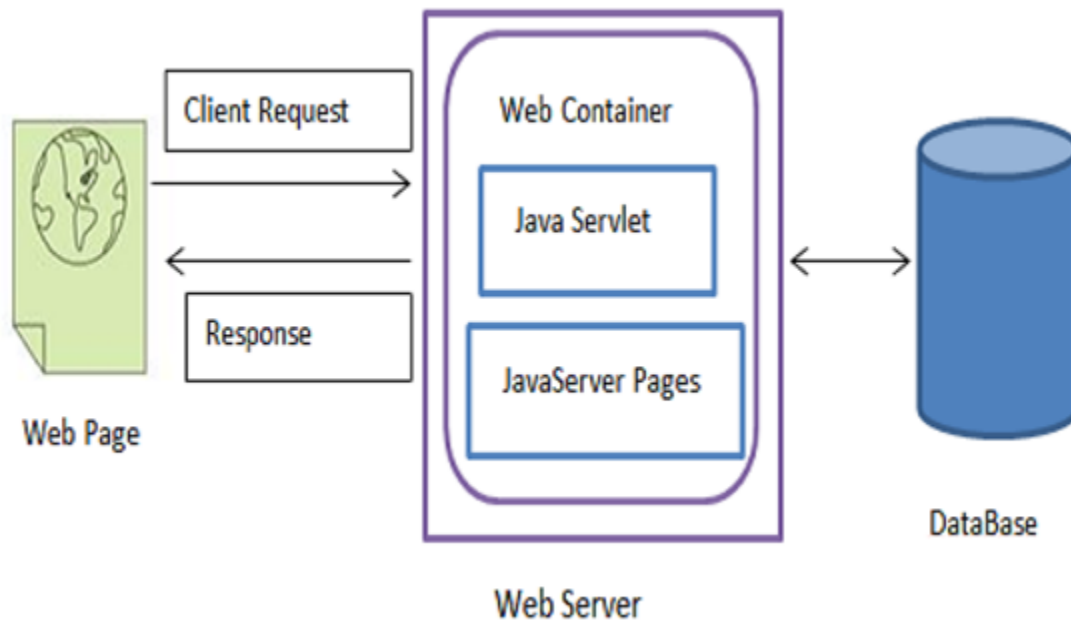


Figure 4.2: Web Server Architecture

A web server is responsible for handling the request coming from the browser. It escalates the request to the different services and then sends back the data/response to the corresponding request coming from the browser.

Bracket IO. Brackets is an open-source editor written in HTML, CSS, and JavaScript with a primary focus on web development (Weber, 2017). It was created by Adobe Systems, licensed under the MIT License, and is currently maintained on GitHub. Brackets is available for cross-platform download on Mac, Windows, and Linux.

Brackets has a major focus on development in JavaScript, CSS and HTML. The latest version release of Brackets is 1.8.

The HTML and JavaScript for the project was written using Brackets IDE. The live preview option of brackets help to maintain a live web environment which is updated after every added line of code either in HTML or JavaScript.

Web browser. A web browser is a software application that serves as a platform for retrieving and presenting the information on the world-wide web. These information resources are stored in the cloud or in physical media and are accessible through a Uniform Resource Locator or commonly called a URL. Web browsers are not only used to view pages in the world-wide internet, but also used to access files hosted in private network and file systems.

The web browser was first invented in 1990 by Tim Berners-Lee (World Wide Web Foundation, 2017). Web browsers have multiple functionality, they do various actions under the hood before presenting the information to the user of web pages. Once the web browser gets the URL address it makes the HTTP or HTTPs connection to the target. The target returns the HTML text and the data in the form of either Extensible Markup Language (XML) or Java Script Object Notation (JSON) and the web browser displayed the information accordingly.

The browser's function can be divided into the following main components:
The User Interface includes the address bar, buttons for various actions like refresh, back and forward, menus to view the history, book mark and the settings of the browser.

The Browser Engine mediates the actions between the UI and the rendering engine.

The Rendering Engine is responsible for displaying the requested content. For example, if the requested content is HTML, the engine parses the HTML and CSS and displays the contents created from these two elements.

Network Service in the browser calls different HTTP requests.

UI Backend is used for drawing the basic widgets like windows, popups and alerts.

JavaScript Interpreter is used to parse and execute the JavaScript alongside the HTML.

Data Storage is the persistence layer. The browser saves various information like history, cookies, etc.

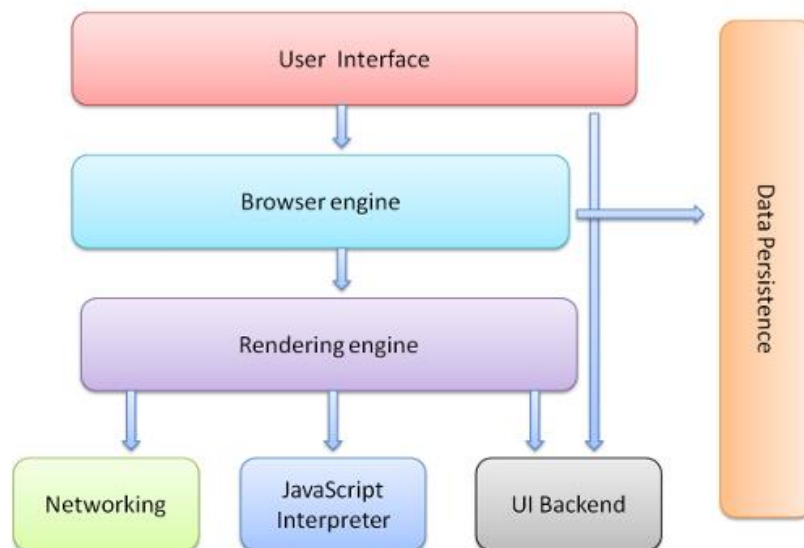


Figure 4.3: Browser Components

Summary

In this chapter, we discussed how this project was implemented using various technologies along with the overview and introduction of those technologies.

Chapter V: Analysis and Result

The main tiers implemented in the application are, calculation of traditional risk score, resource allocation using game theory method and the mixed strategy, which uses both traditional and game theory methods for risk determination and resource allocation.

Home Page

The home page includes a simple UI with the introduction of the research objective. It has a subdivided menu for choosing various options like Traditional Risk Scoring, Game Theory based resource allocation and Mixed Strategy which utilizes both methods for security analysis. These options can be chosen from the easy drop-down menu provided in the UI.

Game Theory Based Risk Computation and Resource Allocation

The security of the computer network has become a challenging task with ever increasing attacks against it. The problem that lies ahead of the management and security administrator is to allocate the resources in accordance with the pitfalls in their network system. While traditional risk analysis can provide them the rough idea of what needs to be protected, an efficient method is imperative to plan the resources to tackle the incoming attacks in more dynamic way. This project is the result of similar study which focussed on modelling a traditional IT risk case into a mixed strategy game and help the security professionals to calculate and place their resources against the intruders with the help of game theory. The added advantage it provides over the traditional risk analysis is that, it helps the administrator to re-allocate resources for the required level of safety for their assets. This research paves a way for converting existing IT risk analysis into a more robust logic based analysis which could prove beneficial in planning real world IT risk analysis and resource allocation.

Risk Computation Menu

Please use this interface to compare and contrast the different risk scoring methods. Please click any of the options available from the drop down menu below.

Select one option ▼



ST. CLOUD STATE
UNIVERSITY.

Developed for the partial fulfillment of starred paper entitled **Improving risk scoring method using game theory** to the Department of **Masters of Science in Information Assurance** at Saint Cloud State University, 2017.

✉ Bikrant Gautam, MSIA, Saint Cloud State University, bikrant.gautam@outlook.com/bgautam1@stcloudstate.edu

Developed using [jQueryUI](#), [jQuery](#) and [fusionChart](#) with their free/public versions.

Figure 5.1: Application Home Page

Risk Computation Menu

Please use this interface to compare and contrast the different risk scoring methods. Please click any of the options available from the drop down menu below.

Select one option ▼
Select one option
Traditional
Game Theory
Mixed



Figure 5.2: Drop-Down Select Menu

Results

Traditional risk score. This option can be chosen by clicking “Traditional” from the drop down menu. On clicking this menu, an additional UI will drop down in the home page giving the brief introduction of what a traditional risk scoring method is. It also provides the UI for providing the data, which we normally use when calculating a traditional risk scoring method.

Traditional ▼

Traditional Method

This method utilizes the method proposed by Micheal Whitman which makes the calculation using the asset value, likelihood, control and the uncertainty property of the asset.
The calculation is done as:

$$(\text{Asset} \times \text{Risk}) - (\text{Asset} \times \text{Risk}) \times \text{Control} + (\text{Asset} \times \text{Risk}) \times \text{Uncertainty}$$

Device name:

Asset Value:

Risk Likelihood:

Uncertainty:

Control:

Figure 5.3: Traditional Risk Score Input UI

After the value is provided, the system will validate the data and if the data are acceptable the “Compute” button calculates the risk scoring. The resulting display is provided intuitively with the risk scoring value in the text along with it in the gauge for

better visualization. The gauge itself has a clean UI to show what region the score lies among the good, neutral and bad.

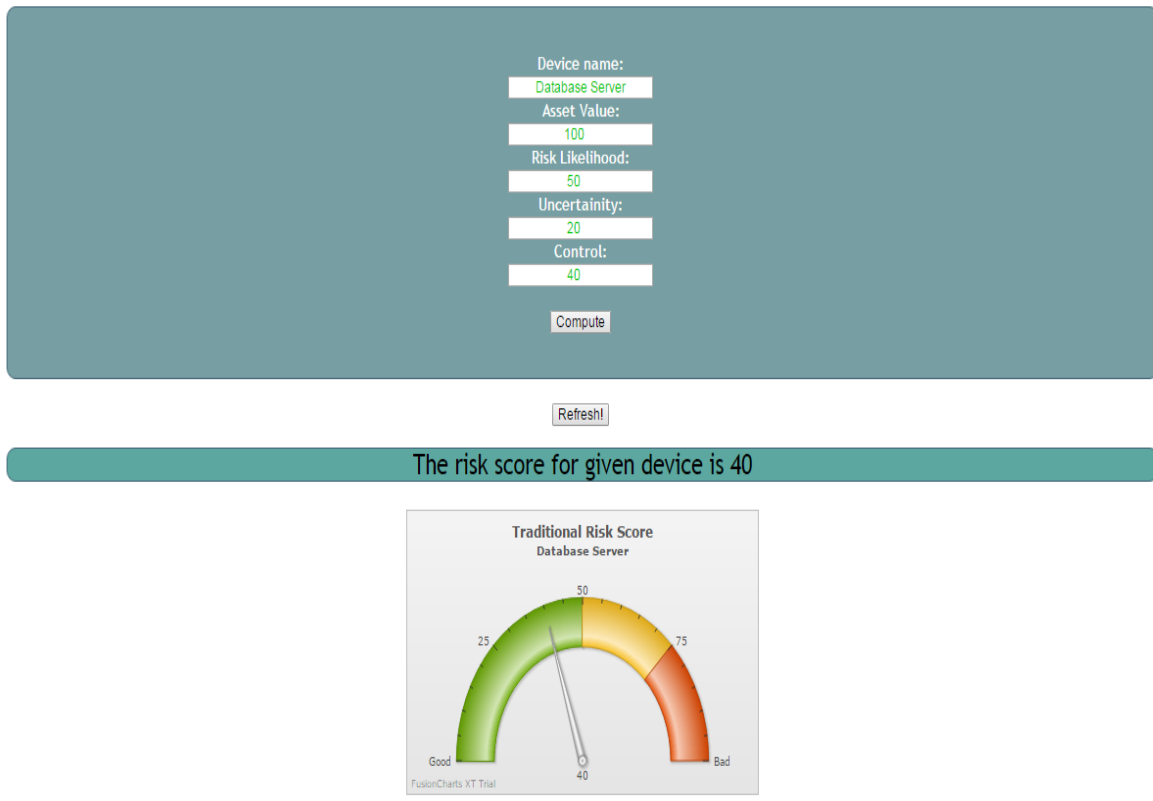


Figure 5.4: Traditional Risk Score Output UI

From Figure 5.4, we can see that the risk score for this device is 40, which is expressed both in text and visually in the gauge chart.

Resource allocation using game theory. This option can be chosen by clicking the “Game Theory” method from the drop down menu. On clicking this menu, an additional UI will drop down in the home page giving the brief introduction of what game theory method is implemented along with the formula. It also provides the UI for providing the data which we normally use while calculating traditional risk scoring

method, making it easier for the user to use, where they don't have to be intimidated with the additional data required.

This method is a competitive based calculation hence it will require more than 1 system to compare and evaluate the resource allocation.

Game Theory ▼

Game Theory Method

This method attempts at implementing a competitive mixed strategy game theory in risk analysis for IT risk management. The objective of this calculation is to set an example of how implementation of game theory could substantially help the security professionals to make decision over the resource allocation for the ever changing vulnerability pertaining to IT field.

The visualization and the charts generated after calculation helps the security experts to make rational decision over the resource allocation.

The game matrix for attacker and defender is constructed as:


	Intruder selects column	
Defender selects row	a_1, a_2	b_1, b_2
	c_1, c_2	d_1, d_2

The attack defense probability for the respective devices are calculated as:

$$P_D = (d_2 - c_2) / [(d_2 - c_2) + (a_2 - b_2)]$$

$$P_A = (d_1 - b_1) / [(d_1 - b_1) + (a_1 - c_1)]$$

Figure 5.5: Game Theory Text



The image shows a web-based input form for a game theory evaluation. It is divided into two columns for 'Device 1' and 'Device 2'. Each column contains five input fields: 'Device name' (text), 'Asset Value' (text), 'Risk Likelihood' (text), 'Uncertainty' (text), and 'Control/Resources' (text). Below these fields are two buttons: 'Compute' and 'Refresh!'.

Device 1	Device 2
Device name: Database Server	Device name: Mail Server
Asset Value: 70	Asset Value: 80
Risk Likelihood: 10	Risk Likelihood: 40
Uncertainty: 30	Uncertainty: 20
Control/Resources: 60	Control/Resources: 40

Compute

Refresh!

Figure 5.6: Game Theory Input UI

The game theory method will first create a risk matrix from the provided data for the competitive game evaluation. After the evaluation, it will provide the defense and the attack probability of each device.

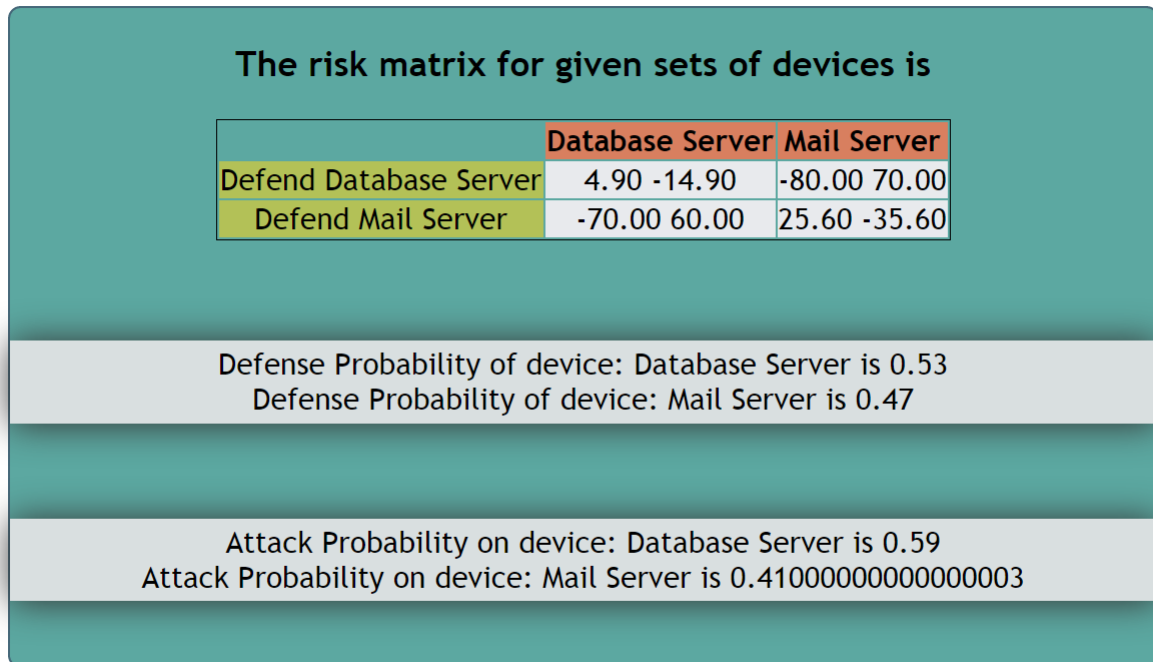


Figure 5.7: Game Theory Risk Matrix

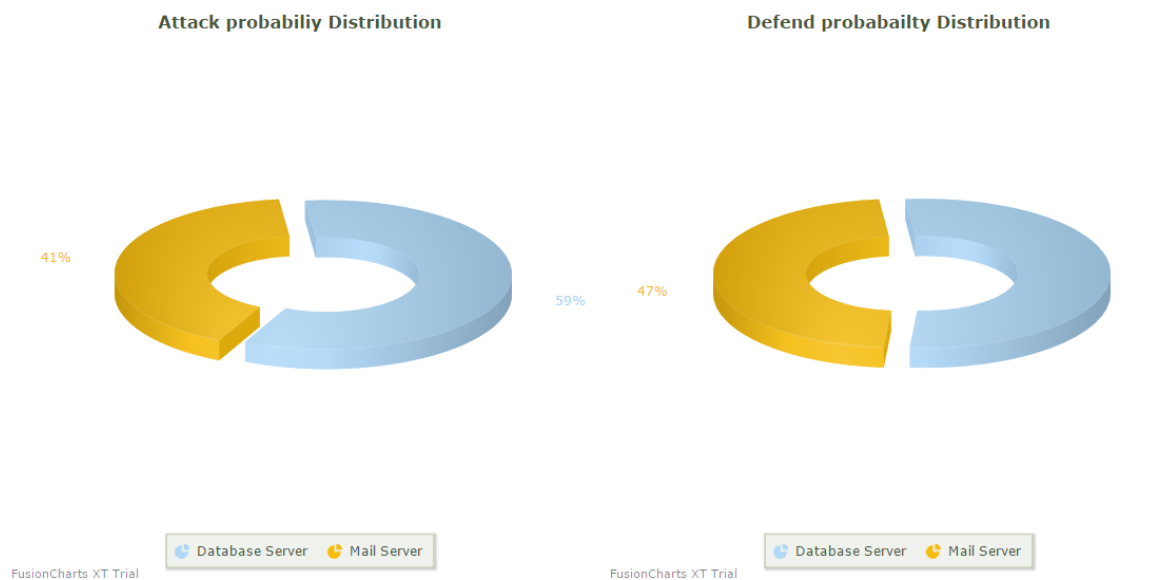


Figure 5.8: Game Theory Probability Distribution

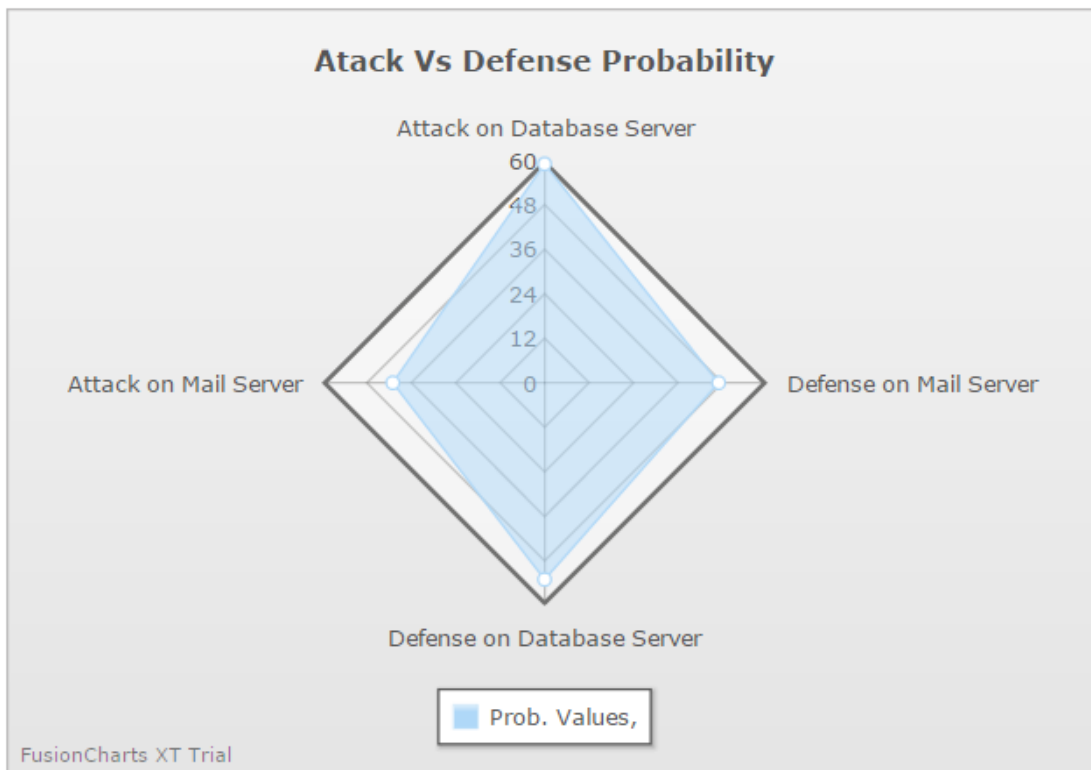


Figure 5.9: Probability Radar Distribution

To maintain the equilibrium between attack and defense the device: **Database Server** should be allocated **59.00%** of the available resources and device: **Mail Server** should be allocated **41.00%** of the available resources.

Figure 5.10: Resource Allocation Suggestion

The end of the report provides the conclusion, which in the given case is:

To maintain the equilibrium between attack and defense the device: Database Server should be allocated 59.00% of the available resources and device: Mail Server should be allocated 41.00% of the available resources.

This method helps to allocate the resources, which is calculated from the existing data provided for the traditional risk scoring method.

Mixed method for risk scoring and resource allocation. This option can be chosen by clicking the “Mixed” method from the drop down menu. On clicking this menu, an additional UI will drop down in the home page giving the brief introduction of what the game theory method is and how it is implemented with the formula. It also provides the UI for providing the data. This method employs both the traditional and game theory method to evaluate the risk and calculate the resources to be applied to the system.

Mix Strategy

This method is the proposed method where the traditional risk scoring is combined with resources allocation as suggested by the computation of competitive game between the attacker and the defender. The numeric values drawn after the computation provides various methods of visualization which will help the security administrator to monitor and defend their perimeter more efficiently.

The calculation for traditional risk scoring is done as:

$$(\text{Asset} \times \text{Risk}) - (\text{Asset} \times \text{Risk}) \times \text{Control} + (\text{Asset} \times \text{Risk}) \times \text{Uncertainty}$$

The game matrix for attacker and defender is constructed as:

	Intruder selects column	
Defender selects row	a_1, a_2	b_1, b_2
	c_1, c_2	d_1, d_2

The attack defense probability for the respective devices are calculated as:

$$P_D = (d_2 - c_2) / [(d_2 - c_2) + (a_2 - b_2)]$$

$$P_A = (d_1 - b_1) / [(d_1 - b_1) + (a_1 - c_1)]$$

Figure 5.11: Mixed Strategy Text

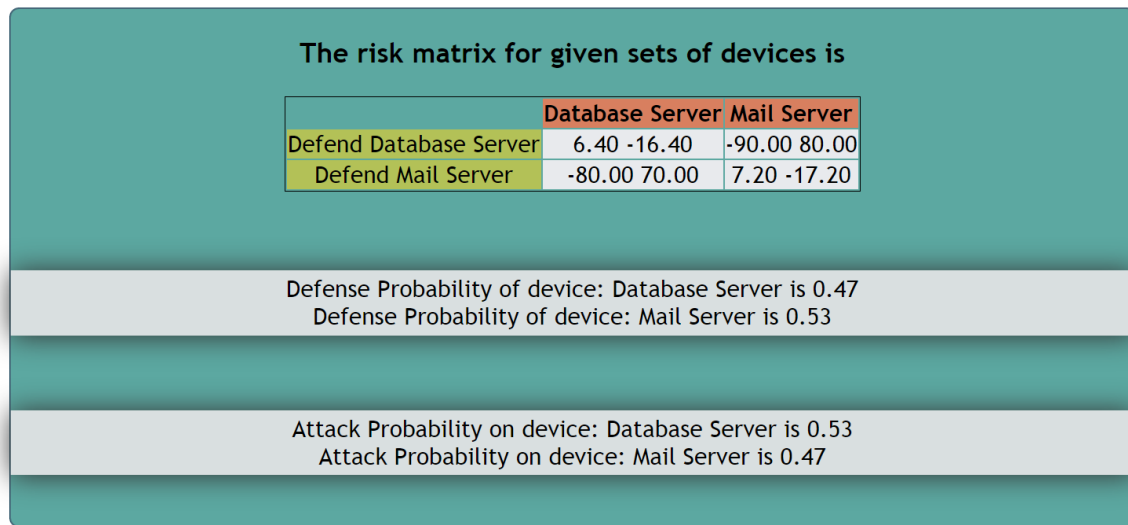


Figure 5.12: Mixed Strategy Risk Matrix

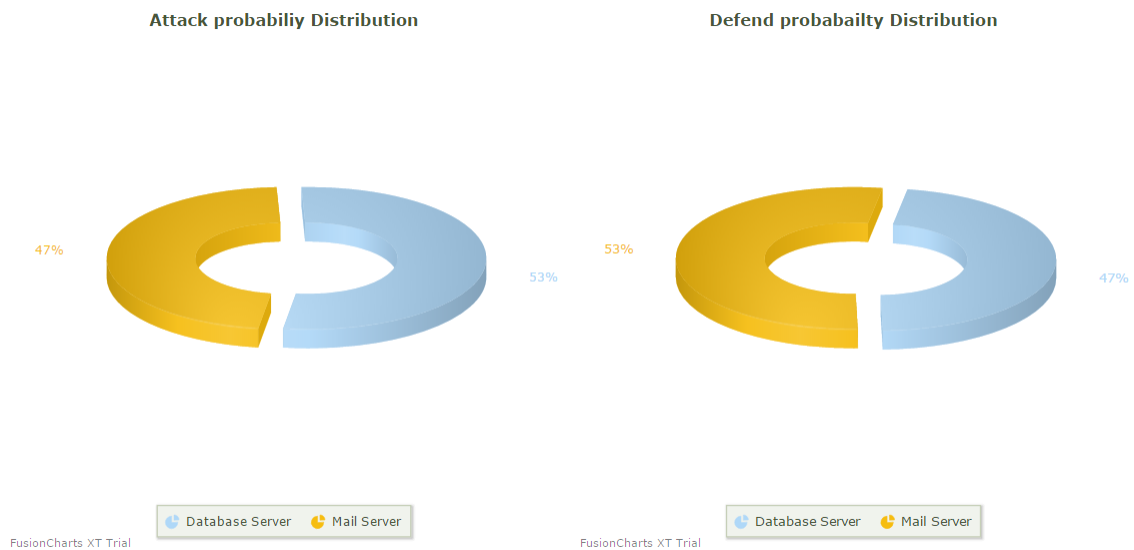


Figure 5.13: Mixed Strategy Probability Distribution

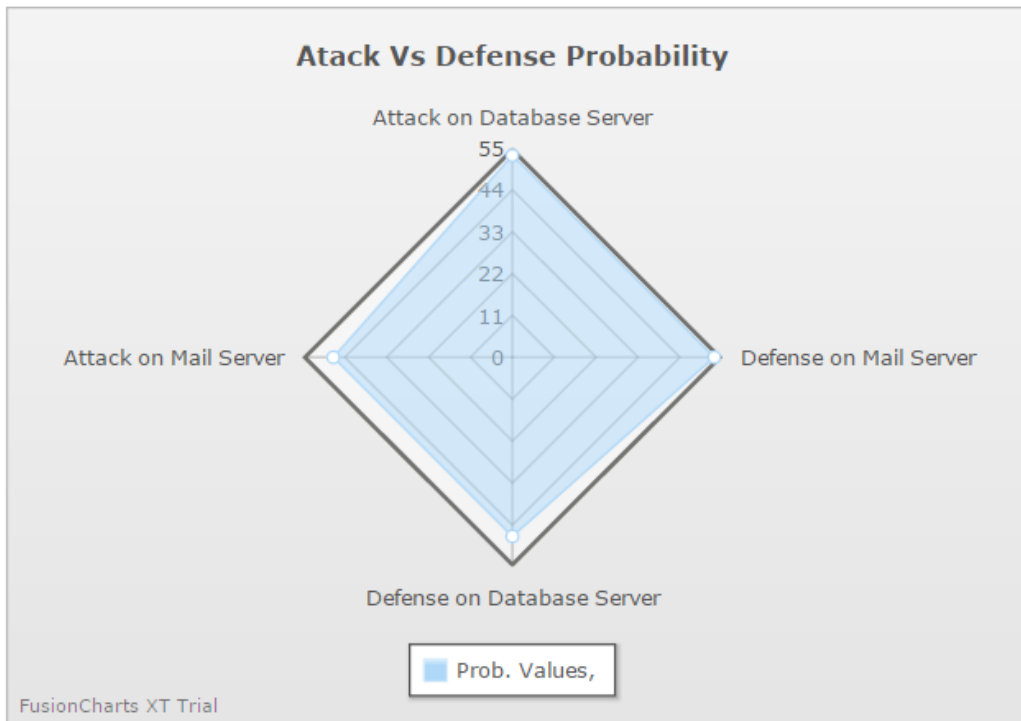


Figure 5.14: Mixed Strategy Radar Distribution

To maintain the equilibrium between attack and defense the device: Database Server should be allocated 53% of the available resources and device: Mail Server should be allocated 47% of the available resources.

Figure 5.15: Equilibrium Probability Distribution

This mixed strategy suggests the probability distribution for the given data to be:

To maintain the equilibrium between attack and defense the device: Database Server should be allocated 53% of the available resources and device: Mail Server should be allocated 47% of the available resources.

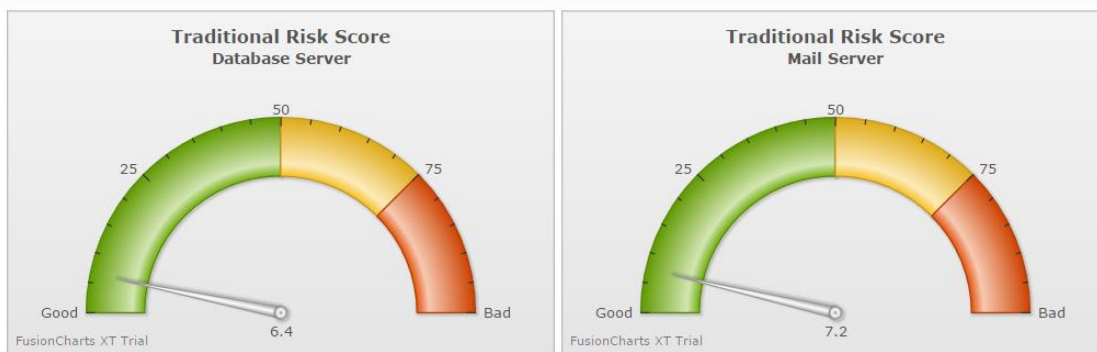


Figure 5.16: Traditional Risk Scoring of Devices

The risk score of device: Database Server is 6.40

The risk score of device: Mail Server is 7.20

Figure 5.17: Traditional Risk Scoring in Text

Conclusion

Combining the two methods: traditional and mixed approach, we can identify and allocate resources for the best outcome with given constraints. We know that the device: Database Server has risk score of 6.40 derived from traditional method, and to maintain in equilibrium, it is provided 53.00% of the available resources. Similarly the device: Mail Server has risk score of 7.20 derived from traditional method, and to maintain in equilibrium, it is provided 47.00% of the available resources.

This approach not only eliminates the uncertainty of risk scoring but also provides dynamic approach to defend resources if any adjustment has to be made on available resources.

Figure 5.18: Conclusion

The conclusion provides the overview of how the resources can be allocated for different devices, for a given input, the conclusion reads as follows:

Combining the two methods: traditional and mixed approach, we can identify and allocate resources for the best outcome with given constraints. We know that the device: Database Server has risk score of 6.40 derived from traditional method, and to maintain in equilibrium, it is provided 53.00% of the available resources. Similarly, the device: Mail Server has risk score of 7.20 derived from traditional method, and to maintain in equilibrium, it is provided 47.00% of

the available resources. This approach not only eliminates the uncertainty of risk scoring but also provides dynamic approach to defend resources if any adjustment must be made on available resources.

Summary

The implementation has taken a simple model of a risk scoring problem and attempted to provide a mathematical solution over the allocation of resources for those risky systems. The implementation has been able to meet the underlying research problem; to assist in resource allocation while calculating the risk for different devices.

Chapter VI: System Design

Data Flow

The data flow diagram, also called a bubble chart is a formal graphical representation of the application or system which describes the input data to the application, the different processes that are done on the input data and the final output data being generated by the application.

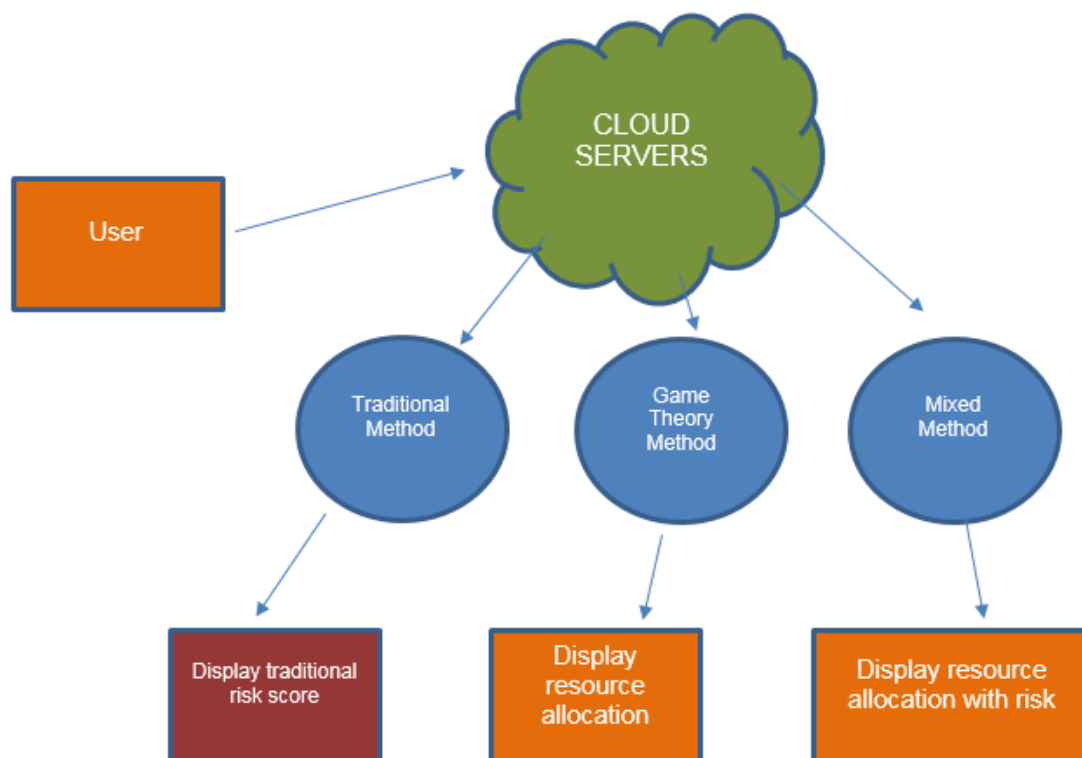


Figure 6.1: Application Data Flow Diagram

- The diagram is represented as a bubble flow chart. It acts as a graphical representation of a system that has input, processing, and output.

- Though being common, it is greatly confused. It is considered as a system procedure that processes. It is also an external entity that interconnects with the system and information flows in the system.
- The diagram is effective in providing an illustration of a system, particularly in abstraction. It is usually portioned into levels that symbolize the rising information flow. It also represents the functional details. It illustrates how information flows in the system and the manner it is changed by a series of transformations (Figure 6.1)

The order of the operation flows from the users to the cloud. The request is then translated into the services available, in this case a traditional risk calculation, game theory risk calculation and mixed method risk calculation. Once the computation is completed, the cloud server sends the response back to the user and provides information for the requested service.

Use Case and Sequence Diagram

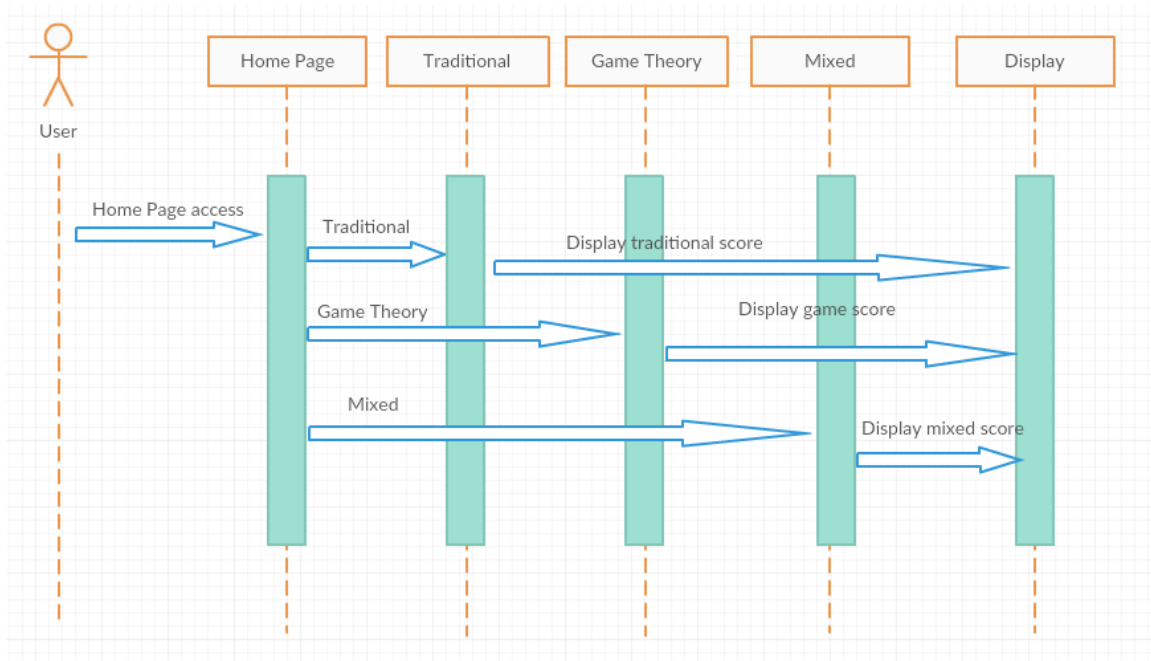


Figure 6.2: Application Sequence Diagram

UML stands for unified modeling language. It is a standardized universal modeling language in object-oriented software engineering. Currently, UML involves two main mechanisms—a Meta-model and notation. In most cases, it uses graphical notations to show the design of software projects. Its goals are:

- Give consumers the overview of the product. It is also an expressive visual modeling language that can construct and come up with and exchange meaningful models.
- Give extensibility and specialty devices to lengthen the fundamental ideas.
- They aim at providing autonomous programming languages and the development procedures.

- They also give an official basis for comprehending the modeling language.
- It encourages the development of the object-oriented device market.
- It also aims at giving support to the design and making of high-level concepts like collaborations.

Use Case Diagram. In UML, this diagram is a form of representation that is described and developed from a use case analysis. Its main objective is to illustrate a graphical representation of functionality that is provided by a system. It shows how the system functions are carried out by a particular actor whose tasks are shown in the Figure 6.2.

The user is the actor and has access to the home page. The home page acts as a facade interface calling the different services within the application. From the user interface, the user can select three of the available options: Traditional, Game Theory and Mixed methods. On selecting one of these options, the appropriate services within these options are called and the values are displayed in the user interface.

The user can then go back to the home page again to select other actions.

Summary

Different ways of developing software using different software designs have been discussed in this chapter. System design is the process of describing and designing the project according to the user requirements. In the data processing industry system design plays a crucial role, it helps in building a modular system by standardizing hardware and software.

Chapter VII: System Testing

Any system that is developed needs to go through a phase of testing. It is a mandatory requirement to find out the bugs and shortcomings in any application. It gives the user an idea about the working of various components and functions both individually and when integrated as a whole. Testing can include several types such as unit testing, integration testing, functional testing, system testing, and white and black box testing. Each of these tests addresses some necessity as required by the application.

Types of Testing

Black box testing. Black box testing is a type of software testing that ignores the internal design of the system and targets only the verification of expected output for a given input. Both positive and negative test case scenarios are validated in this testing. This testing is also termed as a functional testing.

White box testing. White Box testing validates the internal design or application logic of the software component under test on the basis of knowledge of the functionality. This testing is also termed as a Glass Box testing or structural testing. White box testing is often used for verification, whereas Black box testing is used for validation purposes.

Unit testing. Unit testing is done typically at the beginning of the software development phase. Any small unit or software component developed by the programmer must be unit tested. This type of testing is usually done by the programmer instead of the testing team because of the need of in-depth knowledge

of the software unit developed for reliable testing purposes. The programmer needs to test the software component developed with different input values and validate the expected output. It is a form of white-box testing.

Integration testing. Integration testing plays a major role in the application testing. This testing ensures that the application or system is working as expected even after integrating with other modules or external interfaces. The software and hardware interaction used across multiple interfaces is also tested in Integration testing. It is a form of both white box and black box testing. This type of testing is generally performed against client-server frameworks, a distributed environment that interacts over a network.

Different forms of integration techniques are:

- Big-Bang
- Top-down
- Bottom-up
- Mixed (Sandwich)
- Risky-Hardest
- Collaboration integration
- Backbone integration
- Layer integration
- Client- server integration
- Distributed services integration
- High-frequency integration

Functional testing. Functional testing is a testing strategy where the functionality of a software component is tested to work as expected per the defined client or business requirements. This testing usually ignores the internal logic and targets only the expected output. This is a form of black box testing.

System testing. System testing is performed after the complete system is implemented in a typical software project environment. This testing ensures that the entire system is working properly in different environments such as different operating systems or web browsers. This is a form of black box testing where the combination of system parts is tested based on the overall requirements specified.

Stress testing. Stress testing ensures that the system is functioning as expected even in unfavorable situations like heavy system or database load, complex database queries, overloading system capacity, system crash or hang, and power off, etc. The system is tested beyond the testing requirement specifications and focuses mainly on *how the system behaves in failing scenarios. This is a form of black box testing.*

Performance testing. Performance testing plays a vital role in any real-time application where a large number of users are involved. This testing verifies the effectiveness and speed of the system under test. It ensures that required results are generated in an acceptable time period. It is a form of black box testing.

Regression testing. Regression testing is performed to test the application or system after applying some modifications. This testing ensures that the system is

working as expected even after the changes made to it. Usually, different automation tools are used to perform this testing. It is a form of black box testing.

Acceptance testing. Acceptance testing comes into the picture after the software system is completely built and delivered to the customers. This testing is generally performed by the users/customers instead of software testers to ensure that the system delivered is functioning as per the requirements. This testing is also a form of black box testing.

Test Objectives

- Check if all the UI fields are working correctly for valid entries.
- Different pages of the application are linked properly and are in line with the work flow.
- The different messages that notify the user about the status are not misguided.

Features Tested

- Tested all UI fields and elements with different types of valid and invalid data.
- Tested if any duplicates are being allowed or not.
- Tested if the control is being properly moved over the application.
- Tested at the boundaries of the functions wherever applicable.
- Tested if the warning, error and success messages or prompts are displayed properly.
- Tested if all the calculations are executed correctly.

- Tested the application in both positive and negative scenarios.

All the test cases have been passed in all the different scenarios and no defects have been observed.

Summary

This chapter briefly discusses the methods of system testing. System testing is performed to understand the functional specifications of a system and system requirements of the system. The testing is carried out to address issues and test the system under various constraints.

Chapter VIII: Conclusion

In this paper, methods of traditional risk scoring and game theory based resource allocation are studied. With the utilization of both methods, the paper has proposed an advanced analysis, which helps security professionals to improve the risk mitigation process with better resource allocation while calculating the risk score.

A plan has been laid out, investigated, discussed and explained on how to build an application that answers the resource allocation on multiple products present in the system. The implemented application has been thoroughly tested for defects. The result provided by the study now paves a path for further research and could lead to a startup application with the collaboration of more researchers and programmers.

The future expansion of this paper might include a development of a full scale automated application which would map the risks present in various repositories, measure its own available resources and provide a live data visualization or reporting for the security administrators regarding the risk score and resource allocation.

References

- CERT. (2017, January 20). *OCTAVE*. Retrieved from <http://www.cert.org/resilience/products-services/octave/>.
- Cox Jr, L. A. T. (2009). Game theory and risk analysis. *Risk Analysis*, 29(8), 1062-1068.
- FBI. (2016, August 10). *Cyber threats in the 21st century*. Retrieved from https://meeting.afrinic.net/afgwg/presentations/day2/01_02_drccyberthreats.pdf.
- Jesus Rios, A. D. (2012). Adversarial risk analysis for counterterrorism modeling. *Risk Analysis*, 32(5).
- jQuery. (2017, January 23). *jQuery API*. Retrieved from API Documentation: api.jquery.com.
- jQuery. (2017, January 15). *The jQuery project*. Retrieved from jQuery: The write less, do more, JavaScript library: <https://jquery.com/>.
- jQueryUI. (2017, January 20). *jQuery user interface*. Retrieved from <https://jqueryui.com/>.
- JS Foundation. (2016). *Intellectual property policy*. Author.
- Libscore. (2017). January 20). *Library site Cloud*. Retrieved from <http://libscore.com/#libs>.
- Mitra, S. (2017, January 22). *India's rising tech stars*. Retrieved from [forbes.com: http://www.forbes.com/2010/01/07/infosoft-software-orangescape-intelligent-technology-india.html](http://www.forbes.com/2010/01/07/infosoft-software-orangescape-intelligent-technology-india.html).

Nash, P. I. (1999). *Playing it straight: Pure strategy nash. In game theory evolving.*

Princeton University Press.

Osborne, M. S., & Rubinstien, A. R. (1994). *A course in game theory.* London,

England: The MIT Press.

OWASP. (2017, January 20). *OWASP risk rating methodology.* Retrieved from

owasp: https://www.owasp.org/index.php/OWASP_Risk_Rating_Methodology.

/r/Hacking. (2016, August 20). /r/Hacking. Retrieved from reddit:

<https://www.reddit.com/r/hacking/>.

Roy, Sankardas, et al. (2010). A survey of game theory as applied to network

security. System Sciences (HICSS), 2010 43rd Hawaii International Conference on. IEEE, 2010.

Security vulnerabilities. (2016, August). (CVE) Retrieved June 15, 2015, from CVE

Details: <http://www.cvedetails.com/vulnerability-list/year-2016/month-8/June.html>.

Spaniel, W. (2016, September 2). *Game theory 101: The mixed strategy algorithm.*

Retrieved June 17, 2015, from https://www.youtube.com/watch?v=YRECCg7B_L0.

Stanford University. (2016, August 10). *Strategies and tactics for intelligent search.*

(Stanford University) Retrieved June 17, 2015, from <http://web.stanford.edu/~msirota/soco/minimax.html>.

Strassmann, P. A. (2009). *Cyber security for the Department of Defense*. Retrieved from: <http://www.strassmann.com/>.

Weber, H. (2017, January 15). *Adobe launches its open source text editor Brackets out of beta, releases CSS extraction tool*. Retrieved from [venturebeat.com: http://venturebeat.com/2014/11/04/adobe-launches-its-open-source-text-editor-brackets-out-of-beta-releases-css-extraction-tool/](http://venturebeat.com/2014/11/04/adobe-launches-its-open-source-text-editor-brackets-out-of-beta-releases-css-extraction-tool/).

Whiteman, M. E., & Mattord, H. J. (2010). *Management of information security* (3rd ed.). CENGAGE Learning.

World Wide Web Foundation. (2017). *History of the web*. Retrieved from <http://webfoundation.org/about/vision/history-of-the-web/>.

Appendix

1. The complete project is available at the author's Git hub repository at:

<https://github.com/bikos/bikos.github.io>

2. The working example of the project is available at:

<http://bikos.github.io/>