HardnBot

INTELLIGENT SERVER HARDENING SOFTWARE

Project ID: **19_20-J01**

Thesis

H.G.R Aruna Shan

B.Sc. (Hons) Degree in Information Technology

Sri Lankan Institute of Information Technology Sri Lanka

08th May 2020

HardnBot

INTELLIGENT SERVER HARDENING SOFTWARE

Project ID: **19_20-J01**

Thesis

This dissertation is submitted as a partial fulfillment of the Degree of Bachelor of Business Administration Honors Degree Specializing in Marketing Management

Supervisor:

Mr. Amila Senarathne

B.Sc. (Hons) Degree in Information Technology

Sri Lankan Institute of Information Technology Sri Lanka

08th May 2020

DECLARATION

I declare that this is my work and this report does not incorporate without acknowledgment any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of my knowledge and belief, it does not contain any material previously published or written by another person except where the acknowledgment is made in the text.

……..
H.G.R Aruna Shan
IT16099746

The above candidate is carrying out research for the undergraduate Dissertation under my supervision.

Supervisor

…….

Mr. Amila Senarathne

1. Abstract

Compliance-issues are an important issue faced by all organizations. Securing information systems is critical. Organizations should be able to understand the ecosystem and predict attacks. Predicting attacks quantitatively should be part of risk management. The cost impact due to worms, viruses, or other malicious software is significant. This paper proposes a mathematical model to predict the risk score of a compliance issues based on significant factors that influence cyber security. This model also considers the environmental information required. It is generalized and can be customized to the needs of the individual organization. [1] The quantitative model enables compliance risk to be captured in a precise and comparable fashion. The objective of the scoring model is to create a common reference in the marketplace that could enhance incentives for firms to invest and improve their security systems. [2]

Table of Contents

List of Figures

2. Introduction

2.1. Purpose

This report is created to give a clear idea about process of development of the research Project “HardnBot”. and also, by using this document we will explain feature vice details of the project requirements (functional, non-functional vice), how this software works, what are the novelty features of this project, what are the software requirements for best performance and for availability, what are the components we hope to cover in this project, how are we going to cover and fulfil the requirements, so on.

By using this Document anyone can review the process of work flow in the different phases of the project and check whether the project team meets requirement specifications which identified before. This document allows to identify deviation of project outcomes and also accept change in requirements.

This document is created for the research team and the supervisor to get clear idea about functional, non-functional requirements. And also, this document gives clear description about the technologies used to solve research problem which is assigned to our group. Also, this HardnBot report will be guide for researchers interested in implementing this type of tool.

2.2. Overview

We need to emphasize that predicting future events is a different and more difficult problem than detecting current malicious events. For detection, false positives can be very expensive (e.g., a use may not be able to perform their job if an essential software is erroneously recognized as malware); the goal is hence maximizing the true positive ratio while keeping the false positives very low. On the other hand, for prediction the main goal is quite the opposite an enterprise would want to know all the machines that could Be infected, to apply appropriate hardening measures or provide security training to users compared to the detection domain, false positives are more difficult to avoid, but the cost of false positives is lower. Previous works in the prediction field produced more than 20% false positives to predict over95% of the incidents correctly. these numbers are perfectly acceptable for insurance companies. However, the competition in the market raises the bar and asks for lower false positives. Our work aims at meeting this expectation [3].

Recently, several security companies started to incorporate cyber insurance in to their multi-layer cyber security approach, to ensure that the recovery after cyber-attacks is less painful. Due to the high demand for cyber insurance [3].

The market has been steadily growing and putting the insurance companies in a great competition to assess and predict the risk the most accurately.

To solve these types of difficulties and prevent intentional and unintentional human errors, we developed an automated system to solution quantitative relationship between the impact and the parameter of the attack, we are going to discuss the theoretical approach and comparisons regarding previous researches, experiments we conduct, and obtained results based on those experiments. .

2.3. Project scope

HardnBot is a software that has the capability to identify failed operating system compliances of a Unix based servers and classify those failed compliances and use those data to apply industry recommended best practices or organizational required fixes to the Unix based operating system of a server. Throughout this preliminary progress review document, we will explain how failed compliances classification process is going to achieve its goal and steps that needed to be taken.

Under this preliminary progress review document, following components are described.

a) Predict the overall risk score of the server using the classified compliance issues.

b) Display the asset value to be threaten.

HardnBot consist with four main novel components,

- Issue classification
- Risk score prediction
- Intelligent hardening
- Backup and smart rollback

Within those main components there are subcomponents/functionalities and throughout this document, flow of the predicting risk score and identify threatened asset value, these subcomponents will be discussed.

2.4. Background & Literature Survey

i. Research of Information Security Risk Prediction based on Grey Theory and AN [4]

Risk prediction is an important part of the information security system. In accordance with the information security risk assessment process and combination of assets, threat, vulnerability and safety control measures, to strengthen the correlation among these factors and make the prediction results more objective for the target, the authors put forward a model based on the combination of the grey theory and analytic network process (ANP) with information security risk prediction. Establish the weight of each risk assessment element through the analytic network process (ANP) by analyzing interdependency and feedback. Finally, set up systematic risk fuzzy comprehensive calculation to process data and build accurate mathematical model by combining with the risk assessment level.

firstly, the authors grasp the development law of information system through the processing of raw data and the establishment of the grey model, and confirm the preliminary scientific

quantitative prediction for the system's future state; Secondly, use the network analysis method of ANP to compare each independent elements, so that the authors are able to calculate the weight value of each risk factor which affects the system security, reorder the weights, and propose more targeted and objective improvement measures; Finally ,combining with the weight value, to analyze risk objects, the authors obtain fuzzy membership matrix of judgment matrix and build the fuzzy mathematical model, calculate the value of the risk factors comprehensively, and treat it as the guidance, so that reliable guarantee for information system security can be provided. The model realized the grey theory prediction model and was applied in the field of information security, calculate accurate comprehensive weights of various risk factors in information system. In the system, internet elements are interdependent and give feedback to each other, thus combining the theory of fuzzy mathematics, satisfying the requirements of the objectivity and complex of information system, forecasting result is scientific and accurate, instructive significance as well.

ii. Information Security Risk Assessment and Management Method in Computer Networks. [5]

This suggested a method for quantitative information security risk assessment and management in computer networks. This process evaluates an impact and possibility value for specific threats using fuzzy logic and analytic hierarchy process to evaluate. Using fuzzy rules and fuzzy interference system, evaluation vulnerabilities under the uncertainty.

Consider such types of assets - information, host, servers, and telecommunication equipment, IT-services (confidentiality, integrity and availability)

Consider three groups of external socio-political impact, internal impact, and direct financial losses. Most of these are qualitative, thereby they use the analytic hierarchy process for their quantitative evaluation. Evaluate priority weights of information asset regarding to confidentiality, integrity or availability.

Possibility Evaluation for Specific Threat

They suggest a method for quantitative evaluation of threat’s exercising possibility, which is based on the questionnaires. These questionnaires include questions about possibility factors for specific threat and some possible answers to these questions. Answer for every question. After the answering all questions assign number of points and they will assign possibility of the threat.

Vulnerability Evaluation

They suggest a few methods for vulnerability risk assessment. It is based on common Vulnerability Scoring system (CVSS).and they suggest new vulnerability assessment method based on expert judgments, fuzzy production rules and fuzzy logic.

Risk Assessment

They assess the information security risk for specific threat and specific vulnerability by following way.

Risk(Threat)=impact(threat).Possibility(treat).RL

iii. Using a Prediction Model to Manage Cyber Security Threats[6].

Cyber-attack is an attempt to exploit computer systems and networks. Cyber-attacks use malicious codes to alter algorithms, logic, or data. Securing information systems is thus critical. Multiple countermeasures need to be built The CVSS is an industry framework that helps quantify the vulnerability impact. This paper demonstrated a mathematical model to predict the impact of an attack based on significant factors that influence cyber security. Vulnerability and network traffic were selected as the influencing factors to predict CVSS score. Based on the score, the technical analyst can analyze the impact and take necessary preventive actions. This model also considers the environmental information required. It is thus generalized and can be customized to the needs of the individual organization.

TABLE 1: Project data points.

Y CVSS Score	*X*1 Vulnerability	*X*2 Network Traffic
2.1	20	324
5.3	53	623
1.0	15	235
8.0	85	932
2.9	28	438
3.0	25	498
3.8	38	391
1.0	18	132
1.2	16	177
5.9	63	823
4.3	39	579
2.8	30	455
1.1	14	231
4.2	35	725
5.4	51	740
1.9	21	345
2.0	25	432
4.1	37	467
6.2	58	845
1.1	15	111
2.3	22	191
1.2	16	182

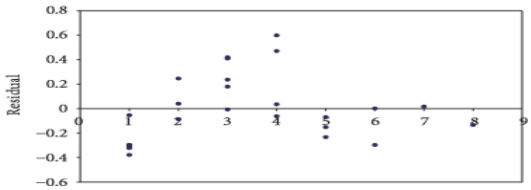

FIGURE 1: Residual plot.

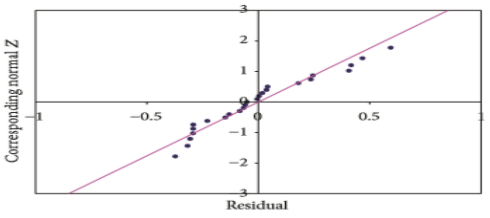

FIGURE 2: Normal probability plot.

Figure 1 :CVSS score prediction

Y is the overall CVSS score. CVSS is the predicted based on the environment and system characteristics of the target application. X1 is the number of vulnerabilities, namely, the total number of vulnerabilities detect by the static and dynamic vulnerability detection tool for target application.X2 is the average input network traffic.

In this regression model, CVSS score predict by the using two variables network traffic and vulnerability. Vulnerability and network traffic have no influence over CVSS score. No mirror pattern can be found (residual plot). Probability plot shown in figure 2is approximately linear. CVSS score is impacted positively both vulnerability and by network traffic.

Predicted CVSS score = intercept + Vulnerability * number of vulnerabilities +network traffic *average input networks

Intercept, vulnerability, network traffic can be calculate using regression equation.

iv. Quantitative Assessment of Cyber Security Risk using Bayesian Network-based model[7].

This paper proposes a quantitative model for assessing cyber security risk in information security. The model can be used to evaluate the security readiness of firms in the marketplace through qualitative and quantitative tools. We propose a Bayesian network methodology that can be used to generate a cyber-security risk score that takes as input a firm's security profile and data breach statistics. The quantitative model enables cyber risk to be captured in a precise and comparable fashion. The objective of the scoring model is to create a common reference in the marketplace that could enhance incentives for firms to invest and improve their security systems. This paper concludes with a demonstration of scoring an intrusion detection network.

The Scoring mechanism determine from questionnaires are generated, the network is complete in both its qualitative and quantitative assessments. The scoring mechanism proceeds with a series of calculations to determine the score of a higher child node and similarly to the resource-driven security score.

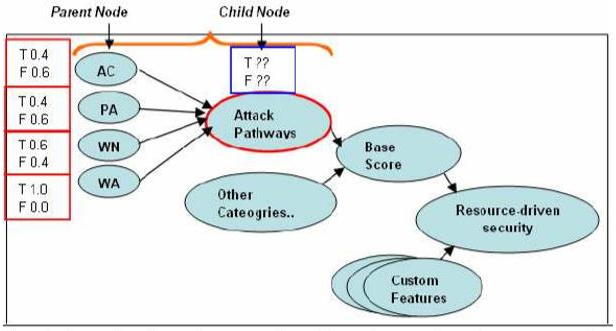

Fig. 8. Intrusion detection network to determine attack pathway prevention sub-score.

Figure 2 : Intrusion detection network prevention pathway

Scoring mechanism perform using Bayesian methodology and probability theorem.

2.5. Research Gap

Until now, there has been little work reviewed in the above researches in specifying or detailing the importance of quantitative risk assessment. Based on researches “Identifying Potentially-ImpactedAreabyVulnerabilitiesinNetworkedSystemsusingCVSS” [8] and research on “A Conditional Probability Computation Method for Vulnerability Exploitation Based on CVSS” [9] were discussed in length and defined as the following.

The Common Vulnerability Scoring algorithm, Bayesian probability theorem and CIS Benchmarks. Using these metrics, I introduce a risk score mechanism to the compliance issue for the server. The risk score prediction for compliances approach hasn’t been reviewed so far. System administrators can be easily reviewed Server current state and generate report for the server current state.

Objectives

2.6. Main Objectives

This research aims to find an effective way to secure the datacenter Red Hat Enterprise Linux version 6, 7, and CentOS version 7, 6 servers and enhance productivity for the customers and employees with a low cost and few human interactions

Here fully automated free open source hardening platform is developed with the capability to detect poor or non-compliant configurations in a system (OS/DB/Application) and applying industry recommended fixes/configurations and secure systems by reducing its surface of vulnerabilities.

This research composes a great business value as it can cover 90% of an Information Systems Audit and hardening process. For internal audits, this software presents both opportunity and responsibility. By helping the organization understand and control risks and identifying opportunities/industry best practices to embrace. This software also will allow the internal audit to position themselves as trusted advisors.

i. Automatically detect uncompliant configurations of the server

   To make these critical servers more secure, this system can automatically perform a dry run feature. With the Dry Run feature, you can execute the playbook without having to make changes on the server [21]. With Ansible Dry Run you can see if the host is getting a configuration changed or not.

ii. Automated Linux server hardening

To prevent intentional and unintentional human errors, the automated system hardening solution explores and highlights the basic security configurations that should be performed to harden the security posture of a default Linux Operating System installation which can provide you with both scheduled or ad-hoc benchmark tests against CIS standard.

iii. Bring industry best practices into the system.

For a particular parameter in the system configuration, there is an industry-recognized value. For example, a password should expire at most in 90 days. A company may not adhere to these standard values; its security policies may not describe them. On such occasions, system administrators may have assigned them with default values. Through our software, we plan to introduce industry-accepted values and configurations into the information systems.

2.7. Specific objectives

i. Reduce manual work

Through this software, we target to reduce the manual effort needed for a complete system hardening. Almost all the parts of the audit and hardening could be performed through this software. Compliance issues are automatically detected. Reports can be generated through this software at the end of the hardening process.

ii. Ease the internal audit

This software is designed in a way that could be easily used by non-technical people. Simple interfaces, pop-up guidelines, and descriptive reports will make the user aware of the functionalities of the software. Even before applying a fix to an issue, the user can read a full detailed report of that issue including its impact and the remediation.

This software package can be handled by a single user. Therefore, the entire system's hardening process can be conducted by a single user. This requires a minimal amount of the organization’s resources. As for the internal audit, this software will cover up to 90% of their work.

3. Methodology

A. Setup connection between HardnBot and the remote server

HardnBot has a separate interface to initiate the connection with a server. This functionality was developed using the SSH.NET library. SH.NET is a Secure Shell (SSH) library for .NET, improved for parallelism and with extensive framework support. It was used to establish a secure shell connection with the server and a new connection to be established, it requires four parameters, IP address, port number, username, and password accordingly.

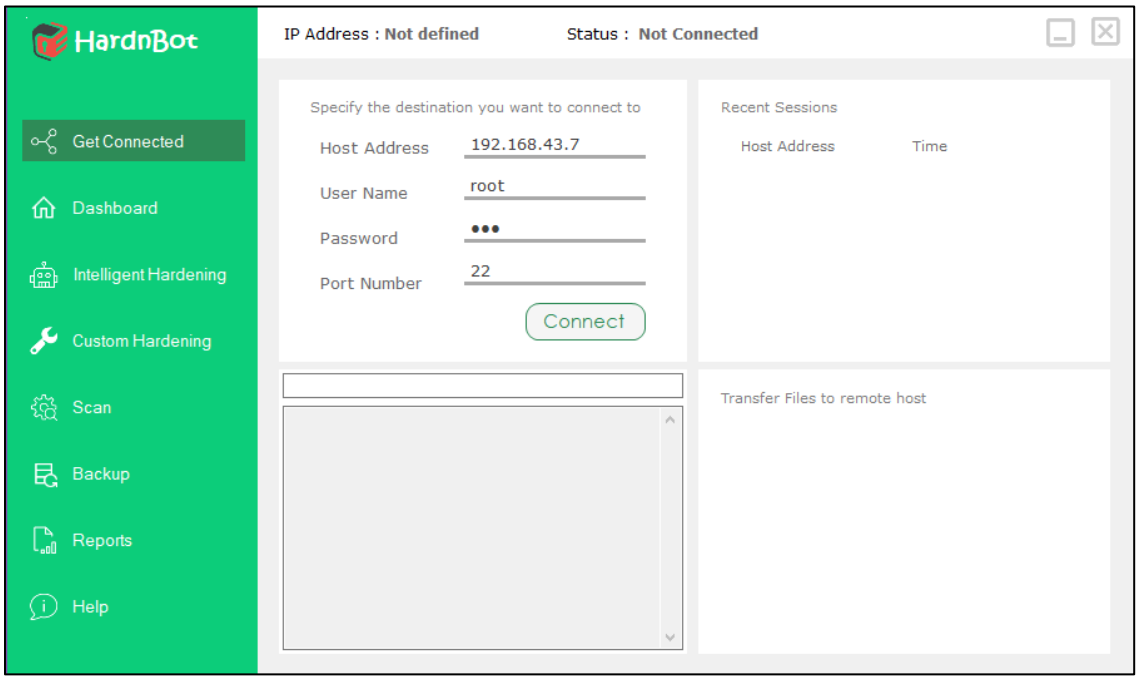

Figure 3 : Get connected interface

```
this.sshClient = new SshClient(ip,Convert.ToInt32(port),username,password);
this.sshClient.ConnectionInfo.Timeout = TimeSpan.FromSeconds(120);
this.sshClient.Connect();
```

After the execution of the above code segment, a secure shell connection will be established with the server.

B. Define report interface.

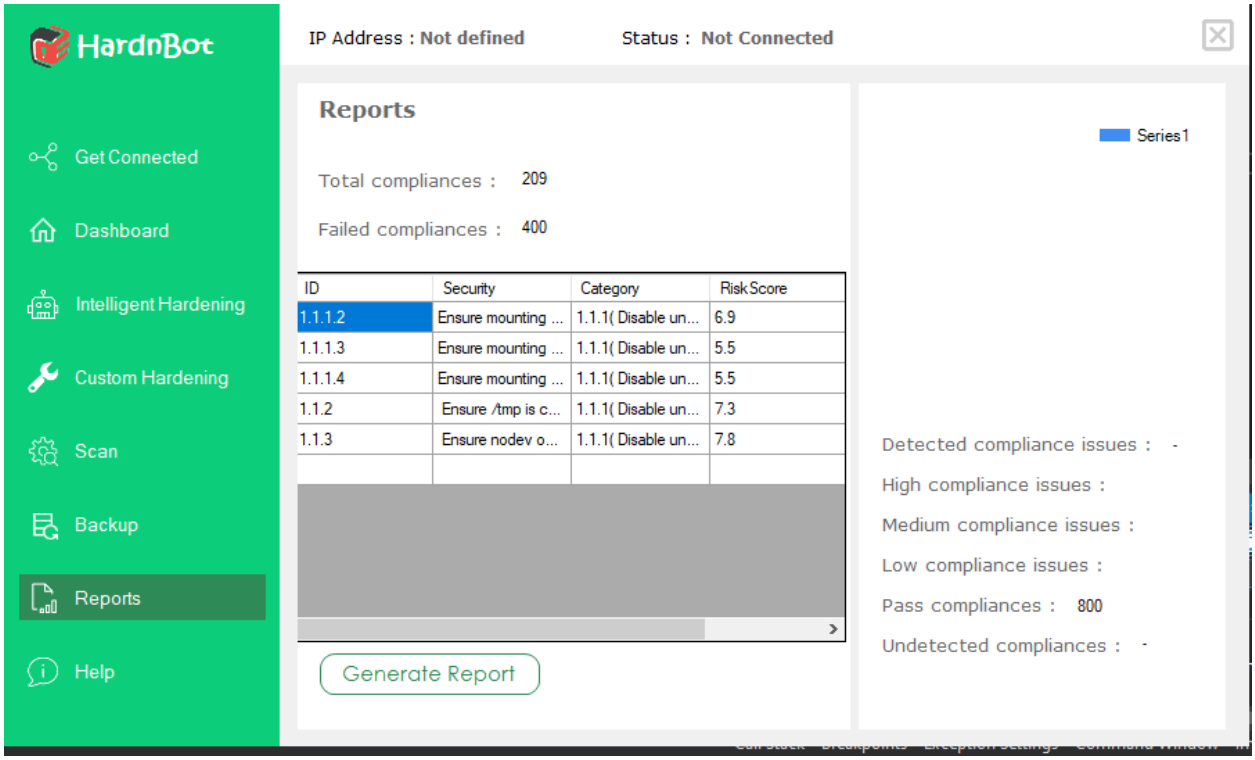

Figure 4 - HardnBot's Report

After the scanning process happen, report window can be open. Inside the report window it will showing ID of the compliance, asset value to be threaten, category of the threaten asset, risk score of the Compliance and the severity level of the compliance. From these details can be generate the report of server current state, two different report will generate, one for before hardening and the after hardening. In the report window showing the result of total compliance, failed compliance, undetected compliance and detected compliance issue.

C. Implement calculation function.

```
            double impactSubScoreBase = 1 - (1 - CImpact) * (1 - IImpact) *(1 - AImpact);

            var exploitabilitySubScore = 8.22 * AVector * AComplexity * PRequred
            *UInteraction;

            Calculater p = new Calculater();
            double impactSub = p.ImpactSubScore(Scope, impactSubScoreBase);

            double BaseScore = p.Score(Scope, impactSub, exploitabilitySubScore);

            double round = p.RoundUp(BaseScore, 1);


            return round;
        }
```

Figure 5 - HardnBot's Calculation function

Calculation function will depend on few variables attack vector, attack complexity, privileges required, user interaction, scope, confidentiality impact availability impact and integrity impact, these variables are generating the risk score for the founded compliances. With the three-impact level generate the “ImpactsubscoreBase” and the “exploitabilitySubScore” will generate with the attack vector, attack complexity, user interaction and privilege required. Inside the “Impact Sub Score” function will generate the impact sub score, there for need to provide two parameters ‘scope’ and ‘impact sub score base’

```
public double ImpactSubScore(string scope, double subScore)
        {
            switch (scope)
            {
                case "U":
                 return 6.42 * subScore;
                case "C":
                 return 7.52 * (subScore - 0.029) - 3.25 * Math.Pow(subScore - 0.02, 15);

                default:
                 throw new ArgumentOutOfRangeException(nameof(scope), "Invalidscope");
            }
        }
```

Figure 5 - HardnBot's Impact sub score function

In figure 4 Inside the ‘ImpactSubScore’ function return the ‘subScore’ , ‘subScore’ is depend on the scope.

```
public double Score(string scope, double impactSub, double exploitSub)
        {
            if (impactSub <= 0)
            {
                return 0;
            }
            switch (scope)
            {
                case "U":
                    return Math.Min(impactSub + exploitSub, 10);
                case "C":
                    return Math.Min(1.08 * (impactSub + exploitSub), 10);
                default:
                    throw new ArgumentOutOfRangeException(nameof(scope), scope,
"Invalidscope");
            }
        }
```

Figure 6- HardnBot's Score function

‘Score’ function will give the ‘Basescore’ of the compliance issues. 'Basescore’ is the risk score of the compliances.

D. Implement Hardnbot table.

	ID	Category	Compliances	Attackvector	Attackcomple...	Privilegerequire	Userinteraction	Scope	Confidentiali...	Integrityimpact	Availabilityim...
▷	1.1.1.1	1.1.1(Disable u...	Ensure mounti...	L	H	H	N	U	H	H	N
	1.1.1.2	1.1.1(Disable u...	Ensure mounti...	L	H	H	R	C	H	H	H
	1.1.1.3	1.1.1(Disable u...	Ensure mounti...	L	L	N	R	U	N	N	H
	1.1.1.4	1.1.1(Disable u...	Ensure mounti...	L	L	N	R	U	N	N	H
	1.1.2	1.1.1(Disable u...	Ensure /tmp is ...	L	L	N	R	U	L	H	H
	1.1.3	1.1.1(Disable u...	Ensure nodev ...	L	L	L	N	U	H	H	H

Figure 7 - HardnBot's table

‘Hardnbot’ table provide the id, asset value to be threaten, category, compliances and calculator function’s parameters, there have no mechanism to generate risk score for the compliance issues there for Hadnbot team gather information of the each and every compliance and split in to the eight parameters, with these parameters we generate the risk score for the scanned falls compliances

E. Implement Hardnbot temporary table.

```
                Conn1.Open();
                String sqlTempT = "CREATE TABLE TempTable" +
    "(ID VARCHAR(10) CONSTRAINT PKeyMyId PRIMARY KEY," +
    "Security VARCHAR(100) NOT " +
    "" +
    "" +
    "NULL, Category VARCHAR(100) NOT NULL, RiskScore decimal(2,1) NOT NULL, Serverity
VARCHAR(50) NOT NULL )";

                SqlCommand cmdTempT = new SqlCommand(sqlTempT, Conn1);

                cmdTempT.ExecuteNonQuery();

                Conn1.Close();
```

Figure 8 - HardnBot's temporary table

Temporary table created with the 'Report' form load, report form interacts with the temporary table this table will delete when the hardening process over.

4. Test Results and Discussions

HardnBot's risk score prediction functions show 95% correctness and accuracy.

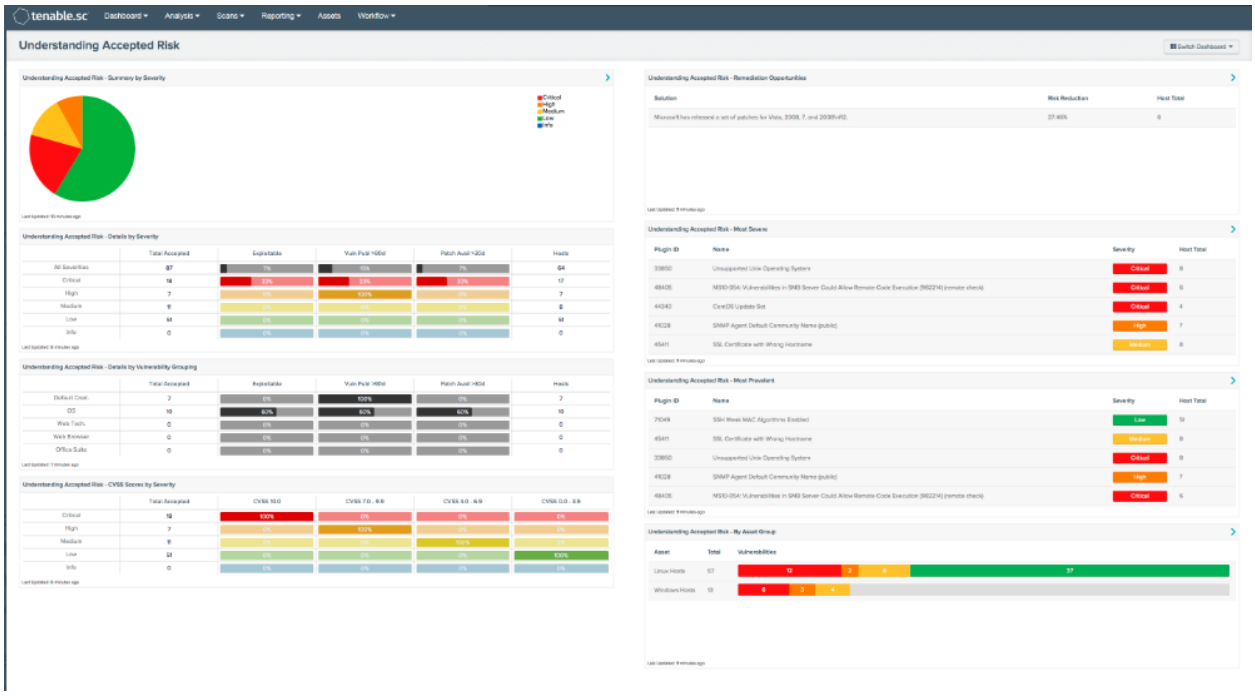

Figure 9 - Nessus Scan results

	ID	Security	Category	RiskScore	Serverity
▷	1.1.1.2	Ensure mounti...	1.1.1(Disable u...	6.9	High
	1.1.1.3	Ensure mounti...	1.1.1(Disable u...	5.5	Medium
	1.1.1.4	Ensure mounti...	1.1.1(Disable u...	5.5	Medium
	1.1.2	Ensure /tmp is ...	1.1.1(Disable u...	7.3	High
	1.1.3	Ensure nodev ...	1.1.1(Disable u...	7.8	High

Figure 10 - HardnBot's Scan results

As in figure 11, we have a Nessus vulnerability scanning report of one of our testing servers (Hosted in a hypervisor) and in figure 12 we have HardnBot's report of the same server.

In figure 11, it shows that Nessus took approximately 2 to 5 minutes to generate risk report this server. By using HardnBot's risk score prediction function the same server can be scanned within 1 to 2 minutes. However, this time may vary according to server content and policies. Nessus is not provide the risk report to the compliance issues in server.

The above details are shows the test results HardnBot and Nessus scanner. By analyzing these data, we can conclude that HardnBot's risk assessment is faster than the Nessus risk assessment function.

However, the accuracy/correctness of HardnBot's risk report are not 100% accurate when comparing with risk report but in the range of 80% - 90%.

.

5. References

[1] Jaganathan, V., Cherurveettil, P. and Muthu Sivashanmugam, P., 2020. Using A Prediction Model To Manage Cyber Security Threats.

[2] "Quantitative assessment of cyber security risk using bayesian network-based model - IEEEConferencePublication", Ieeexplore.ieee.org,2020.[Online].Available:https://ieeexplore.ieee.org/a abstract/document/5166177. [Accessed: 13- May- 2020].

[3] L. Bilge and Y. Han, RiskTeller:PredictingtheRiskofCyberIncidents, 1st ed. United state, 2017

[4] Qian Yu, Yongjun Shen, "Research of Information Security Risk Prediction," in IEEE, 2016.

[5] I. V. Anikin, "Information Security Risk Assessment and," in IEEE, 2015.

[6] Venkatesh Jaganathan, Priyesh Cherurveettil, Premapriya Muthu Sivashanmugam, "Using a Prediction Model to Manage Cyber Security Threats," in ResearchGate, 2015.

[7] Sheung Yin Kevin Mo, Peter A. Beling, Kenneth G. Crowther, "Quantitative Assessment of Cyber Security Risk using Bayesian," in IEEE, 2009.

[8] IdentifyingPotentially-ImpactedAreaby VulnerabilitiesinNetworkedSystemsusingCVSS, 1st ed. Tokiyo, 2010.

[9] A Conditional Probability Computation Method for Vulnerability Exploitation Based on CVSS. China, 2017.