HardnBot

Intelligent Server Hardening Software

Project ID: 19_20-J01

Preliminary Progress Review Report

Bachelor of Science (Hons) in Information Technology

Specialization in Cyber Security

Sri Lanka Institute of Information Technology

16th September 2019

HardnBot

Intelligent Server Hardening Software

Project ID: 19_20-J01

Preliminary Progress Review Report

Supervisor: Mr. Amila Senarathne

Bachelor of Science (Hons) in Information Technology

Specialization in Cyber Security

Sri Lanka Institute of Information Technology

Sri Lanka

16th September 2019

DECLARATION

We declare that this is our own work and this Preliminary Progress Review (PPR) report does not incorporate without acknowledgement any material previously submitted for a Degree or Diploma in any other University or institute of higher learning and to the best of our knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

……..

R.M.B.B. Rathnayake

(IT16054400)

……..

G.G.L. Anjula

(IT16022416)

……..

W.M.K.M.W. Wijekoon

(IT16167742)

……..

Aruna S.H.G.R

(IT16099746)

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

Supervisor

…….

Mr. Amila Senarathne

Table of Contents

List of figures……...5

1. Introduction……6

1.1.Purpose……..……6

1.2.Scope……6

1.3.Overview……..……7

2. Statement of Work……..…..……....8

2.1.Background…….…….…….…….8

2.2.Identification and Significance of the Problem……..…….……...15

2.3.Technical Objectives……..17

3. Research Methodology……24

4. Test Data and Analysis……26

5. Anticipated benefits……26

6. Project Schedule……...28

7. Specified Deliverables……..29

8. References……30

List of Figures

Figure 1: Project Schedule 28

1. Introduction

1.1.Purpose

Preliminary progress review (PPR) is created to provide a clear understanding of the research project and its research components. And, by this document, we expect to explain in depth about the main features and their novelty and what are the hardware or software requirements needed in order to reach the full potential of the completed software (HardnBot) and how are we planning to achieve our final goal.

By using this preliminary progress review report, anyone can review the process of the workflow in each phase of the project and this document allows to identify any deviations of project outcomes.

This document is created for the research team and the supervisors to get a clear idea about functional and non-functional requirements, how to reach our research goal, and what technologies and tools to be used.

1.2.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 all the tasks are going to achieve their goals and steps that needed to be taken.

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

a) Implement scans to detect compliance issues.
b) Automatically harden the system to classified compliance issues.
c) Bring industry best practices into system.
d) Implement backup function
e) Implement intelligence rollback function
f) Setup SSH connection to servers through secure VPN tunnel
g) Scan for compliance failures.
h) Classify them using machine learning approaches into severity levels as Critical high Medium and Low.
i) Predict the overall risk score of the server using the classified compliance issues.
j) 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

1.3. Overview

HardnBot – An automated unix server hardening software platform that has the capability to detect and classify poor/non/failed compliance issues and configurations of an unix server and applying hardening solutions according to the industry recommended best practices or organizational recommended best practices. HardnBot also has the capability of predicting the overall rick score by considering main and all available risk factors and if there is any abnormal behavior after the hardening process, HardnBot has the capability to roll back the modification to an accepted previous level.

Normally these hardening processes will be done by either network administrators, system administrators, outsourced professionals or server custodians by manually running scripts, commands and queries against the server and it will roughly take more than six hours to completely harden a single server in the infrastructure. Probability of a misconfiguration occurrence is higher because hardening will carry out with human interaction. Scenarios where a misconfiguration occurs, it may be hard to detect those issues since some issues cannot be identified via an error message. So, in a scenario like that, system administrators, server custodian or network administrators need to go back to the initial state of the server operating system via a backup image which will be a time-consuming task. By introducing a special feature/function for improve backup and rollback tasks, fully backup to the initial state will not be necessary. Rather than that the software itself can automatically identify the misconfigured spot and rollback only to that point. And by classifying these compliance issues, HardnBot can provide required data to the prediction algorithms as well as to the intelligent hardening features. Also classified (categorized) data can be displayed as a summary to reporting purpose.

HardnBot uses scanning mechanisms to find and identify every compliance issues and misconfiguration of a unix server and this goal is achieved by executing a structured and configured script against the target server. And it will extract all the failed compliances and format them to a CSV formatted file for later use in classification, prediction and hardening functions.

After all the classification is done, a clean classified (categorized) and summarized failed compliance issues will be forwarded to the risk score prediction algorithm as a set of risk factors and later as a parameter to perform the hardening process.

2. Statement of Work

2.1.Background

Massive amounts of data are created daily across the planet. By 2021, the annual global Internet Protocol (IP) traffic is predicted to reach 3.3 zettabytes. To match this huge data environment, the data center industry is anticipating unprecedented growth. Data is the most precious asset in data centers. Data centers require abilities to ensure data service works properly; many technologies are used in data centers to achieve this goal. Data centers are supported to run 24/7/365 without interruption. Planned or unplanned downtime can cause business users serious damage.
Most data centers include Linux servers; Ubuntu Server, Red Hat Enterprise Linux and CentOS. Datacenter includes about 200 live servers it is very difficult to do the operating system hardening manually.
There is no fully automated hardening platform implemented yet. Even the network administrators plan to do the Hardening processes manually it might take more than 6 hours for the complete harden processes for only for one and may contain lots of paperwork. By the way there can be mistakes and faults in the hardening process that can affect the live Servers. Sometimes a company may have to hire an external party to perform the hardening or they may have to outsource their systems to external organizations to assess their compliance. This will cost them in advance. Cost of maintaining compliance and governance may higher than the risks associated with these systems. And, there could be risks in outsourcing critical information systems.

A research conducted by Prowse D. in 2010 on “OS Hardening and Virtualization” describes how to perform OS hardening using OS security audit method. In order to perform OS hardening, as a first step they perform vulnerability assessment over windows, then perform the security audit and fully analysis system logs. Further they explained how important it is to perform periodic security audit over an OS in order to track vulnerabilities and take relevant countermeasures. As benefit of doing OS hardening, they printed out how it helps to reduce the risk, improve the performance, eliminates vulnerable entry points and mitigate security risks. As this paper status OS hardening can be done using techniques such as program clean-up, service packs, patch management, group policies, see templates and configuration baselines. Further for more user friendliness, operating systems like windows provides facilities to prioritize vulnerabilities as high, medium and low. To strengthen the security of OS, they discussed manual technology as well as semi-automated terms under manual techniques. Preparing checklist for security parameters, reviewing security configuration aspects, manually set security configuration and explaining OS as per configuration parameter included. In semi-automated way they are using scripts for audit such as .bat, .ps, set security configuration using script, exploiting OS scripted pay load. In discussion they showed how important it is to perform periodic audits to identify security issues, prioritize those and treat in order to mitigate risk over operating systems [1].

A research conducted by Sanjay Garg on “Network Scanning & Vulnerability Assessment with Report Generation” ha reviewed two of the well-known open source scanners NMAP (Network Mapper) & OpenVAS (Open Vulnerability Assessment System).

They show us how to incorporate these two scanners into a decently outlined GUI and give reliable information. Effectiveness of network scanning and vulnerability testing depends on scanners and processes to scan the network and its devices. Sometimes, use of these tools can lead to device or information being compromised or destroyed by exploits. Different implementations & tools of network scanning have different kinds of outputs. But these outputs are typically heterogeneous.

The network scanner created in this thesis carries out the scanning through the network identifying the active hosts and guessing the operating system of the remote hosts and the programs installed in the remote hosts.

In addition to identifying active hosts, you could find open ports and list the services that run on the host. The exploration of additional vulnerabilities is done by comparing the information obtained from a network scan with a database of vulnerability signatures to generate a list of vulnerabilities that are likely to be present in the network. In this dissertation, the characteristics of the new tool are explored. In other words, network mapping, vulnerabilities, and configuration failures in the network are displayed in various formats. In addition, an easy approach is defined to reduce the duration of vulnerability exploration [2].

AAP is not just a vulnerability scanner. It is a complete system auditing tool that can perform variety of security tasks on a system.

Zhe Wang, Jin Zeng, Tao Lv, Bin Shi and Bo Li published a research paper in 2016 on "A Remote Backup Approach for Virtual Machine Images". In there they were talking about virtual backup on a cloud storage. When we are considering cloud computing, virtualization is playing a major role, because of hosting several applications and services in virtual machines (VM) which were hosted in cloud environments. Security become a prior requirement in virtualized applications. In this research, mainly focused area is high availability issue in virtual machines. LiveRB (Live remote backup) is the proposed remote backup approach. The purpose of the Live RB is to save the running state of the VM in an online manner known as "Live Migration". This backup process will happen the background of the hosted cloud applications of the VM and is transparent to them. A virtual block device will be designed and will be used to cache of I/O Operations in memory, in order to save the incremental virtual disk data.
LiveRB will be implemented on KVM virtualization platform in order to evaluate effectiveness and efficiency using a set of comprehensive experiments. These experiments are all related to Cloud Computing and the security issues that come along with this and the key points considered in order to have successful cloud computing are security, availability & fault tolerance. The commonly used solution to handle Fault Tolerance & High Availability is using snapshots or checkpoints that periodically record the states of the software for backup and rollback the cloud applications to the previously backup up state. This procedure will be carried out when encountering Failures or Errors of the original system.
Most currently existing VMs stop the VM to take snapshots. Some VMs need to be shut down too take snapshots which this affects the ability to provide the service/ result in abnormal cloud application behavior. Some VMs suspend the current process and save the current

progress onto local disks to be transferred onto remote servers later which sometimes result in data loss if a hardware failure is encountered.
The above issues can be resolved using the Live RB since it works by not stopping the VM to do the backup process. Results of this process indicate that Live RB can be used on a VM to do the backup task from VM onto a Remote Server with only a slight reduction in performance [3].

In this research, it described about method that used to back up a virtual machine, but when it comes to our research area we have to consider about live server. Therefore, no need of care about any virtual machines, but when we are talking about remote backup approach used in here, that was Live Remote Back up, so we can consider about this technique when we are dealing with our problem. L. Farinetti and P. L. Montessoro published a research paper in 1993 which named "An Adaptive Technique for Dynamic Rollback in Concurrent Event-Driven Fault Simulation". In here it is discussing about automatic rollback based on an adaptive mechanism which is including advanced network/system status recording system. Time can be any time, that mean before changing of a system or after changing a system this status recording can be apply. Main feature of this research is user can define the rate for maximum acceptable level for rollback. This approach takes the average time to minimum level, that means very short time of rollback process.
To come up with proposed technique, researchers were used existing methods such as incremental backups, journal files, checkpoints, rollback, roll forward which were found on different applications, different operating systems as well as different databases. Mainly the status of the network/system is record on disk and run for negative time period to analyze previous status. If needed user can run for a positive time period as well. Those time periods are for compare with current status of the network/system. To make it happens above approaches need some fine tunes as well [4].

According to rollback techniques used by those researchers we identified some techniques and requirement that should be in our system too. For this part of the software it is necessary to detect abnormal behaviors of the server after the hardening process is done. For that we need to record system status after the hardening process. Then compare with the previous status, that means system status before the hardening process, but in our approach, there are predefine models for compare with current system status. Apart from that rollback mechanism is going to adopt from this research. That are the things we are going to take from this research.

Ning Lu and Yongmin Zhao published a research paper in 2018 which named "Research and Implementation of Data Storage Backup". In this research, researchers were tried to discuss about features of a reliable and secure backup and types of backup. With use of applications which were depend on big data, the usage of data storage backups was became more important. Therefore, the methods used to backup should be more flexible and can be able to ensure of security and reliability of backup contents and backup and restore should be in a convenient manner. There are several backup methods such as data backup, system backup, application backup etc. The backup contents are guaranteed to be confidential, complete and effective.

There are several specific performances in a backup,

i) Backup should be upgradable, capacity expansion
ii) Management without affecting other application in the system
iii) Implement a backup storage system combining SAN (storage area networks) and NAS (network attached storage) storage networks.
iv) Provide several backup methods such as data backup, system backup, application backup,
v) Backup contents should be secure and restore operations should be done in a convenient manner.

System backup

Reefers to the backup of the end-point operating system, server operating system and other systems. In here core files and system's registry are backed up as a data. In a matter of system crash or operation mistaken the backup can be restoring to the previous state.

Virtual tape library

Virtual tape library (VTL) considered as a world's leading modern technology to create a backup system. It can rapidly backup and rapidly recover a system that we want to backup. Main feature is no manual intervention of this technology. VTL storage media is a SATA disk and its data transfer rate are 150MS/s. That means approximately it takes 10 seconds for transferring 1.5GB data to the backup storage [5].

In our research one of a main goal is to reduce the overall hardening time. For achieving that task, we should have to minimize the overall backup time to some acceptable level using speed backup mechanism. In this point we are going to use technique which is described in this research known as virtual tape library. If we can adopt this mechanism overall hardening time will reduce averagely by 8 hours to 4 hours.

Teruaki Sakata, Teppei Hirotsu, Hiromichi Yamada and Takeshi Kataoka published a research paper in 2007 which named "A Cost-effective Dependable Microcontroller Architecture with Instruction-level Rollback for Soft Error Recovery". This tool is developed for detect soft errors using electronic design automation (EDU) which generates optimized soft error detecting logic circuits for flip-flops. When a soft error is detected that signal goes to a developed rollback control module (RCM). That RCM will reset the CPU and restores the CPU's register file from a backup register file using a rollback program guidance. After that CPU will able to restart from the state which is before the soft error occurred. In here researchers were developed another two modules called error reset module (ERM) that can restore the RCM from soft errors and error correction module (ECM) that corrects errors in RAM after error detection with no delay overhead. In above mentioned soft error means, which are random transient errors. Those errors are the main cause of failures in microcontrollers which include reversal of a memory element's bit data due to factors such as alpha rays in a package, neutron strike and noise of the environment [6].

D. R. Avresky and M. I. Marinov published a research paper in 2011 which named “Machine Learning Techniques for Predicting Web Server Anomalies”. The basic idea between servers on the web is to provide requests made by the client through the web using different transmission methods such as Services. Businesses relying on these services require the web servers to have reliability, availability and security in order to provide constant quality in the service provided. This document describes the quality ensured in these services.

The assumption made for this problem is mainly due to Resource Starvation. Resource Starvation is when a process that functions in Concurrent Computing is unendingly denied the necessary resource to continue & process the rest of its work. Resource Starvation is measured by the response time taken to cater requests under artificial workloads while collecting data on other resource parameters. The research provides proof that these recordings gathered from different artificial workloads can be applied to real world entities as well
Machine Learning is used to monitor & correlate the high response time, and this is done by observing the system data. The goal of this analysis is to resolve issues of this variety in Web Servers, Operating Systems or in VM (virtual machine) Rejuvenation.
Based on the statistics provided by the Internet World Statistics, we could clearly notice a rapid rise QoS (quality of service) Internet Service Usage users and this gave several companies & industries to exist in the current world. The below listed out Companies/ Industries who gets affected by these figures since their prime business is offering Internet QoS,

- Cloud Computing
- Data Storage
- Hosting Providers
- Content Delivery
- Application Performance Management & other

Due to this high demand and dependence on network QoS, it is important for a service to be aware of its own deteriorating quality. Currently there are several self-monitoring network products that ensure that the QoS of services offered through the internet. The goal of this this research is to increase this area.
The benefits taken from this research can be applied to other areas as well and they have been listed down below,

- Proactive Software Rejuvenation
- Web Server Workload Balancing
- Web Server Performance Testing

  Other… [7]

In this research mainly focused about detecting anomalies on a web server using machine learning technique. Hence, we are not going to use machine learning techniques for detecting anomalies in a server this research is not a to good feed for us.

Ratsameetip Wita and Yunyong Teng-Amnuay of Department of Computer Engineering, Chulalongkorn University, Bangkok, Thailand published a research paper in 2005 on *"Vulnerability Profile for Linux".* In this research, they talk about profiling identified vulnerabilities according to the CVE score of them. In their classification scheme, they consider four types of classification schemes namely,

1. Confidentiality violation
2. Integrity violation
3. Availability violation
4. System compromised

If a confidentiality violation occurs, it allows an attack to directly steal information from the system. Integrity violations allows an attack to directly change the information passing through the system. Availability violation results an attack that limit the genuine access to a genuine user (human or machine), Denial of service attacks (DOS) can be taken as an example. According to their research system compromised attacks gives the attacker the privilege to access the system in four different levels such as: run an arbitrary code, elevate privilege, account break-in, and finally root break in which can be the worst-case scenario.
Furthermore, these classifications are again grouped according to the severity level.

Damage Type	Severity Level		
	High	Medium	Low
Confidentiality	- Disclosure of information and system configuration in root/super user level	- Disclosure of system information and configuration in user level	- Disclosure of some no relevant information.
integrity	- Information and system configuration changed in root/super user level	-Information changed in user level	- Non-relevant information changed in another user level
availability	-Whole system crash or unavailable	- Some services unavailable -System temporary unavailable	- Some services temporary slow down with flooding
System compromised	-Root break-in -Account break-in	- Privilege gain in some domain	- Run arbitrary code by

	-Run arbitrary code by root/super user privilege	- Run arbitrary code by user privilege	another user privilege

[8]

Here they are classifying vulnerabilities of the system / server, but we are going to develop a software toolkit that is capable of identify and classify failed compliance issues of a server operating system.

Bo Shuai, Haifeng Li, Mengjun Li, Quan Zhang and Chaojing Tang of School of Electronic Science and Engineering National University of Defense Technology Changsha, Hunan, P. R. China published a research paper in 2013 on *"Automatic Classification for Vulnerability Based on Machine Learning*".* This research paper is based on vulnerability classification using machine learning methods based on LDA model and SVM. Word location information is introduced in to LDA model called WL-LDA (Weighted location LDA), which is somewhat better than typical language processing algorithms because it generate outcomes from vector space on themes other than on word, and a multi-class classifier called HT-SVM (Huffman Tree SVM) is developed that it could make a faster and more stable classification on the vulnerabilities [9].

The main idea of this research is that they classify vulnerabilities with new models based on existing LDA and SVM models and obtain more accurate and more effective outcome.

Vijaya MS, Jamuna KS and Karpagavalli S of GR Govindarajulu School of Applied Computer Technology Coimbatore, India published a research paper in 2009 on *"Password Strength Prediction using Supervised Machine Learning Techniques".* This research is targeted on the password strength of a system and predict password strength of a system whether it's a strong password or a weak password using supervised machine learning techniques such as classification (discrete) and regression (continuous).

Here the password strength prediction is modeled as classification task and supervised machine learning techniques were used as mentioned above. In this research they used some common classification models such as,

1. Decision tree classifier
2. Multilayer Perception
3. Naïve Bayes Classifier
4. Support Vector Machine (SVM)

For testing and selecting the best classification model for the performance of the task. After some performed tests, they identified Support Vector Machine (SVM) as the most suitable model for this task [10].

Kai Liu, Yun Zhou, Qingyong Wang and Xianqiang Zhu of Science and Technology on Information Systems Engineering Laboratory National University of Defense Technology Changsha, China published a research paper in 2019 on *"Vulnerability Severity Prediction with Deep Neural Network".* Multiple deep learning methods for vulnerability text classification evaluation are proposed in this research paper. Three kinds of deep neural networks,

1. CNN,
2. LSTM,
3. TextRCNN

And one kind of traditional machine learning method

1. XGBoost

are used. Here all parameters tuned via experiments to improve the accuracy of the task. However, in this research they said that the deep neural network methods evaluate vulnerability risk levels better, compared with traditional machine learning methods but it costs more time to train. This research says that they scored 93.95% accuracy level when training the model [11].

According to these literature reviews, there are some ideas and methods that could be consider in order to achieve our primary goal in classification the compliance issues of a server operating system.

2.2.Identification and significance of the problem

When performing a hardening process, system administrators, network administrators, server custodians or outsourced expertise need to ensure security of the operating system that runs on a server (in our case its unix based servers) database and application because a single mistake can affect the whole production line which the server is in. Even though there are many scanning tools that has the capability of scan a server and identify compliance failures along with the solutions, the solutions are going to apply with a human interaction. So, the probability of mistake occurrence is higher. And manual hardening process consume much more resources such as time, human, cost likewise. As a solution for the lack of resources, organizations are tending to consider about hiring external professionals and assets to perform the hardening task. In a scenario like this, internal critical classified information might have a possible chance to expose via outsourced professionals intentionally or unintentionally and leave the server in a critical position of been hacked or data leaked. And there is a compulsory requirement of the root (administrator) access to the server terminal in order to scan and

perform operating system hardening. Also, the server needs to be temporarily out of live production, because operating system hardening cannot be performed while the server is in live production environment. So, when a critical day-to-day serving server is downed for maybe more than six hours to perform operating system hardening, it will be a critical impact on the organization's day-to-day business activities.

As an example, for all these above described scenarios, we can consider an operating system hardening process of a card server of a commercial bank that provides customers with all kind of credit and debit card services. Since the hardening is done by outsourced professionals and a down time is required, all servers information are available to outsourced personals that include customer card details as well as the server details such as the root password and also because of the downtime required, legitimate customer services will be down. So, in a case like this it will be a critical situation to the organization. Likewise, there are more drawbacks to a manual operating system hardening process.

To solve these types of difficulties and prevent intentional and unintentional human errors, we are going to implement a software platform (HardnBot) which automate the server operating system hardening process and which has the capability of detect failed compliances, classify them based on their criticality/severity levels and apply industry recommended best fixes for them via CIS benchmarks or via organizational requirements. HardnBot is also consists with an automated hardening function that harden a server according to classified (categorized) compliance issues and a rollback function that rollback to a point so that it will maintain and improve the productivity and integrity of the server.

In order to identify failed compliances, we will design a shell-based script that has the capability of executing and collecting all compliance failures in an unix system. After that, there will be a classification segment where the identified failed compliances will be classified (categorized) according to a severity level which measured by the impact criticality. Later these data will used to identify the level of hardening and as a risk factor parameter for the risk score prediction algorithm.

2.3.Technical objectives

Main Programming Language: C#

Almost all operating systems support C# language. C# is a general-purpose, multi-paradigm programming language including strong typing, lexically scoped, imperative, declarative, functional, generic, object-oriented, and component-oriented programming disciplines. And, C# contains with lots of supporting third party libraries. Therefore, C# will be used as the main programming language to implement this software.

IDE: Visual Studio

Microsoft Visual Studio is an integrated development environment from Microsoft. It is used to develop computer programs, as well as websites, web apps, web services and mobile apps. It mainly supports for C# development and contains lots of features that improve programming experience. And because of the inbuild functions and configurations, we are going to use Microsoft visual studio as our development IDE.

Modern UI Development: Bunifu DLL

Bunifu framework is recognized for innovative interface development in C# and C++ environments. Bunifu includes a range of colors and animated controls. It also includes scripts that allows to apply animated effects into the interfaces. Therefore, we will use this framework in designing our interfaces.

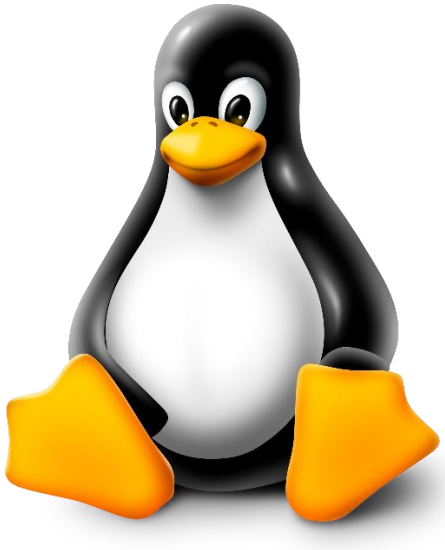
Virtual Environment: VMware Workstation

VMware Workstation is the industry standard for running multiple operating systems as virtual machines (VMs) on a single Linux or Windows PC. As the need we have to run several server-based operating systems to test our product, we will use VMware workstation for run multiple operating systems and observe the product outcomes.

Virtual Environment: Oracle VM VirtualBox

Oracle VM VirtualBox is a free and open-source hosted hypervisor for x86 visualization. In some cases where we have any difficulties to work with VMware, we will consider using Oracle VM VirtualBox as an alternative.

Operating System: Linux



Since our project is targeting linux based systems, we will implement hardening for a series of linux based operating systems as listed below.

{RedHat Enterprise Linux 7, RedHat Enterprise Linux 8, CentOS 7}

Operating System: Windows

Microsoft Windows operating systems are the most popular and wide used operating systems in the world. We will use Microsoft windows operating systems to run our IDEs and implement our software. Below listed versions will be used.

{Windows 8.1, Windows 7, Windows 10 version 1903}

Programming Language: Python

Python is an interpreted, high-level, general-purpose programming language. In our research, python will be mainly used for data training and machine learning purposes. Python includes a lot of mathematical libraries and data manipulation libraries which will be useful for our data training processes. We use following inbuild and third-party libraries. Also, we will use latest stable python version, at the time of developing (currently 3.7.4).

{Numpy, Scipy, Scikit-learn, Theano, TensorFlow, Keras, PyTorch, Pandas, Matplotlib, NLPs}

Automation Tool: Ansible

Ansible is an open-source software provisioning, configuration management, and application-deployment tool. It runs on many Unix-like systems and can configure both Unix-like systems as well as Microsoft Windows. It includes its own declarative language to describe system configuration. We will use ansible to achieve our expected automation hardening process.

Live Coding platform: Jupyter Notebook

The Jupyter Notebook is an open-source web application that allows to create and share documents that contain live code, equations, visualizations and narrative text. Using Jupyter Notebook we can perform data cleaning and transformation, numerical simulation, statistical modeling, data visualization and all machine learning tasks. To use Jupyter notebook, we will use following techniques.

{Anaconda distribution of Jupyter, Hard installed versions of Jupyter}

GPU Based Machine Learning libraries: CUDA Toolkit

The NVIDIA CUDA Toolkit provides a development environment for creating high performance GPU-accelerated applications. The toolkit includes GPU-accelerated libraries, debugging and optimization tools, a C/C++ compiler and a runtime library to deploy our application. CUDA will mainly be used for performing text recognition and for natural language processing tasks.

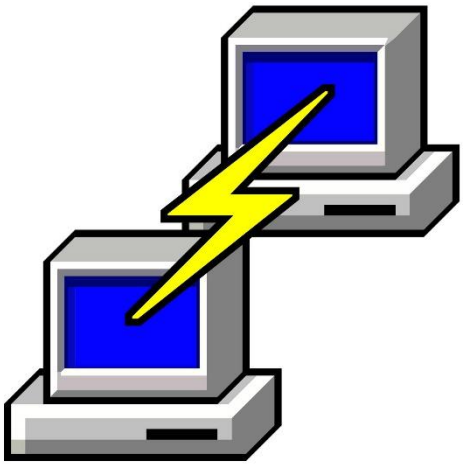
System monitoring tool: Nagios

Nagios, also known as Nagios Core, is a free and open source computer-software application that monitors systems, networks and infrastructure. Nagios offers monitoring and alerting services for servers, switches, applications and services. This tool will be used to monitor our servers for any misbehaviors after the hardening.

Script Language: Shell

Shell script A shell script is a list of commands (a program) designed to be run by the unix shell. We will use shell script for scanning purposes and other types of command executions in linux servers.

Terminal emulator: Putty



PuTTY is a free and open-source terminal emulator, serial console and network file transfer application. It supports several network protocols, including SCP, SSH, Telnet, rlogin, and raw socket connection. We will use putty for testing purposes and to create connections between servers.

Hardware: External GPU installed PC

A system with an externally installed GPU will be used for machine learning purposes.

3. Research Methodology

1. **Implement scans to detect compliance issues.**

   HardnBot perform scans to identify misconfigurations in operating systems. The software will run scripts to retrieve configurations for relevant locations and record them. These scans will retrieve existing system configurations and those configurations will be compared and detect compliance issues. Then these compliance issues will be presented to the user in a well formatted way.

   In order to identify compliance issues HardnBot will scan configurations of followings,

   1. Install Updates, Patches and Additional Security Software 2. OS Services 3. Special Purpose Services 4. Network Configuration and Firewalls 5. Logging and Auditing 6. System Access, Authentication and Authorization 7. User Accounts and Environment 8. Warning Banners 9. System Maintenance

   Scanning for compliance issues is carried out by a well-prepared shell script.

2. **Automatically harden the system to classified compliance issues.**

   After identifying compliance issues, to remediate the issues a script is automatically generated. This script is executed to systems by Ansible automation tool.

   The default hardening and remediation to identified compliance issues is carried by configurations in CIS benchmarks. System owner doesn't have to access Unix environment anymore. The hardening configurations will be displayed to users with appropriate parameters.

3. **Bring industry best practices into system.**

   For a 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; their security policies may not describe them. In 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.

   The hardening configurations will be displayed to users with default CIS benchmarks parameters. Editable GUIs are designed to users to customize these configurations according to industry requirements. Then these parameterized configurations should be passed to Ansible play books through HardnBot.

4. **Implement backup function.**

   Existing system configurations should be taken into backups as a precaution if the new configurations failed. For this purpose, scripts will be designed for each OS/DB components. These scripts will generate a single backup file which can be used to restore in case of applied fixes failed.

5. **Implement Intelligent Rollback function.**

   Rollback function will take place after the hardening process is done. This function basically depends on the abnormal behaviors of the server which are appearing after the hardening process. In here there are several pre-define behavior models to detect those kinds of abnormal behaviors. For that purpose, we create our own models as well as we can use existing models in various tools such as NAGIOS for this task. If those models detect any anomaly regarding to server services, there is a rollback script to run for establishing previous status (backup) of the server and it will automatically run.

6. **Setup SSH connection to servers through secure VPN tunnel**

   When connecting to the server which is going to harden, it should be done in secure manner. For fulfill this requirement SSH connection through a VPN tunnel is the better way. All the processes should be done through this secure SSH session.

7. **Retrieve failed operating system compliances and classify.**

   Classification is done by a trained machine learning model. Machine learning model will be selected after thoroughly test and analyze a set of classification models. Modifications and combinations may perform to the model in order to achieve the best accuracy levels. Supervised machine learning techniques and algorithms will be used to train our datasets.

   Following tools and technologies will used to train the model,

   1. Python 3 – programming language that contains required libraries to perform machine learning tasks.
   2. Jupyter Notebook – tool that use for maintaining live python code and visualize the behavior of the data sets and algorithms.
   3. TensorFlow – free and open-source software library for dataflow and differentiable programming across a range of task.
   4. Anaconda – a free and open-source distribution of Python programming language for scientific computing and machine learning.

8. **Predict the overall risk score of the server.**

   Using the classified compliance issues an equation/algorithm is developed to predict the overall risk score of the server. Using the likelihood of impact and the probability of occurrence, predict the overall risk score of the server. The scoring mechanism proceeds

with a series of calculations to determine the score. For that Probability function and Bayesian model used to determine the likelihood of impact and probability of occurrence.

9. **Display the asset value to be threaten**

   Consider about Information, host, servers, and telecommunication equipment, IT-services (confidentiality, integrity and availability) and identify asset and their values After that identify the asset value to be threatened and display these assets.

4. Test data & analysis

HardnBot's scanning procedures are done by executing scripts against a server and found compliance failures are classify using a trained machine learning classification models in help with text processing models. We will refer previous researches on vulnerability classification and text classification to gain required knowledge to train the datasets with relevant parameters. Online available genuine data sets will be used to train models. Cloud based python servers maybe used to test and retrain our models.

CIS (Center for Information Security) is a well-recognized and well reputed organization on information security and they have published many accurate and reliable reports on operating system compliances. In their website, there are many benchmark tests that they had published for many operating systems and application packages. We will refer the required and relevant reports and will gain required data to improve, test and optimize the datasets and algorithms that are going to use for model training.

We will use VMware workstation or Oracle VM Virtual Box to setup several virtual unix based server operating systems (CentOS, RHEL) and then we will test the scanning function and trained machine learning models for accuracy.

5. Anticipated benefits

HardnBot has a great commercial value because of the functionalities that it provides for users as well as organizations. Our primary goal is to automate an entire server hardening process and with the unique functionalities, HardnBot's user will gain following main benefits.

- **Fully in-depth server scanning specifically for OS compliances**
  With this functionality, HardnBot can scan thoroughly to find any compliance failures and any configuration errors.

- **Failed compliance classification and summarize**
  After collecting data from the scan, HardnBot's machine learning algorithms will provide a detailed list of compliances failures and misconfigurations with their criticality level. By using these, HardnBot will provides user with a detail percentage

level of severity existence (For example: 10% Critical, 20% High, 30% Medium, 40% Low). With this information server custodians can get an overview idea about the severity level of the server.

- **Risk Score**
  By this functionality, an overall detailed risk score will be displayed to the custodian/user so that the organization can get a rough idea about server's possibility of compromise, likelihood and loss.

- **Intelligent Hardening**
  HardnBot' s hardening function is a unique function because it will harden the system for an acceptance level as required by the organization. Obviously 0% risk acceptance is not possible but in this function it's algorithms will use pre-classified compliance failure data and harden with respect to the severity levels.

- **Smart rollback functions**
  Using these functions, HardnBot will monitor applied fixes on the run as well as the server behavior parallelly and identify any misbehaviors and go back to that point where the misbehavior took place and rollback to the default or previously backed-up settings.

Besides those benefits, users will not require any down time to perform hardening because HardnBot's capability of performing stepwise hardening, stepwise error checking, stepwise backup and stepwise rollback. So, each fix will be monitored and having that, these functions will help to improve security operation center's (SOC) productivity and accuracy.

6. Project Schedule

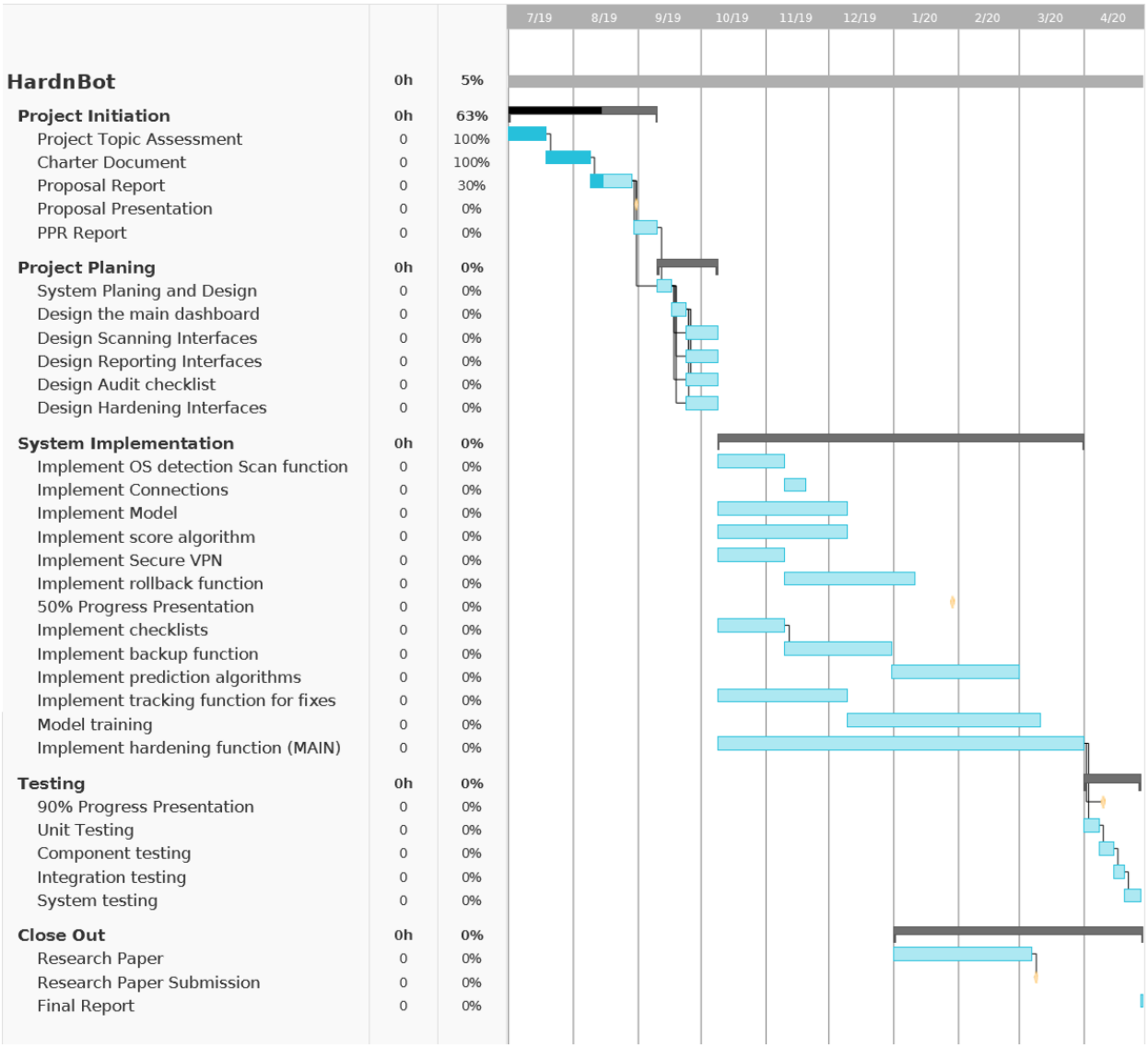
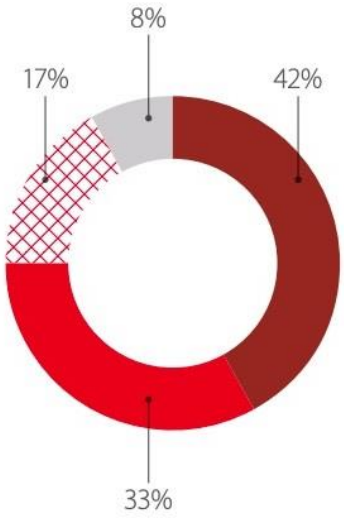


Figure 1: Project Schedule

7. Specified deliverables

- A well formatted compliance and misconfigurations list that classified according to severity levels.

- Total percentage value of compliance and misconfigurations count



- A software that will require very less human interaction

8. References

[1] Prowse, D. (2010). CompTIA Security+ Cert Guide: OS Hardening and Virtualization. [eBook] PEARSON. Available at: http://www.pearsonitcertification.com/articles/article.aspx?p=1667482 [Accessed 12 Mar. 2019].

[2] Sanjay Garg (2014). Network Scanning & Vulnerability Assessment with Report Generation. [online] NIRMA University. Available at: https://www.researchgate.net/publication/263779662 [Accessed 12 Mar. 2019].

[3] Zhe Wang, Jin Zeng, Tao Lv Bin Shi, Bo Li, "A Remote Backup Approach for Virtual Machine Images," in IEEE, 2016.

[4] L. Farinetti, P. L. Montessoro, "An Adaptive Technique for Dynamic Rollback in Concurrent EventDriven Fault Simulation," in IEEE, 1993.

[5] Ning Lu, Yongmin Zhao, "Research and Implementation of Data Storage Backup," in IEEE, 2018.

[6] Teruaki Sakata, Teppei Hirotsu, Hiromichi Yamada, Takeshi Kataoka, "A Cost-effective Dependable Microcontroller Architecture with Instruction-level Rollback for Soft Error Recovery," in IEEE, 2007.

[7] D. R. Avresky, M. I. Marinov, "Machine Learning Techniques for Predicting Web Server Anomalies," in IEEE, 2011.

[8] Y. T.-A. Ratsameetip Wita, "Vulnerability Profile for Linux," 25 April 2005. [Online]. Available: https://ieeexplore.ieee.org/document/1423610. [Accessed August 2019].

[9] H. L. M. L. Q. Z. C. T. Bo Shuai, "Automatic Classification for Vulnerability Based on Machine Learning," 27 January 2014. [Online]. Available: https://ieeexplore.ieee.org/document/6720316. [Accessed 2019].

[10] J. K. K. S. Vijaya MS, "Password Strength Prediction using Supervised Machine Learning Techniques," 12 January 2010. [Online]. Available: https://ieeexplore.ieee.org/document/5376606. [Accessed 2019].

[11] Y. Z. Q. W. X. Z. Kai Liu, "Vulnerability Severity Prediction With Deep Neural Network," 19 August 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8802851. [Accessed 2019].