CYbersecurity Oriented Training Environment and Exercises - CYOTEE

A Thesis

Presented in Partial Fulfillment of the Requirements for the

Degree of Master of Science

with a

Major in Computer Science

in the

College of Graduate Studies

University of Idaho

by

Animesh Pattanayak

Major Professor: Daniel Conte de Leon, Ph.D.

Committee Members: Jim Alves-Foss, Ph.D.; Michael Haney, Ph.D.

Department Chair: Terry Soule, Ph.D.

December 2019

Authorization to Submit Thesis

This thesis of Animesh Pattanayak, submitted for the degree of Master of Science with a Major in Computer Science and titled "CYbersecurity Oriented Training Environment and Exercises - CYOTEE," has been reviewed in final form. Permission, as indicated by the signatures and dates below is now granted to submit final copies for the College of Graduate Studies for approval.

Advisor:	Daniel Conte de Leon, Ph.D.	Date
Committee Members:	Jim Alves-Foss, Ph.D.	Date
	Michael Haney, Ph.D.	Date
Department Chair:	Terry Soule, Ph.D.	Date

Abstract

Cyber defense competitions provide students with a hands-on, real world opportunity to learn, practice, and perform the tasks which they will be expected to complete in the workplace. Cyber defense competitions include red-blue exercises (attack-defend), capture the flag competitions, level based challenges, and environment configuration type activities. The availability of current training material for cyber defense competitions is limited, especially when considering the subset of materials which are targeted for cyber defense competitions, freely available, and modifiable for extended use. CYbersecurity Oriented Training Environment and Exercises (CYOTEE) is a project intended to fill the need for preparatory material targeted for traditional cyber defense competitions. The project is freely available with all content downloadable from the project GitHub repository:
`https://github.com/CenterForSecureAndDependableSystems/CYOTEE`.

The specific contributions described in this thesis are nine laboratory exercises: six hands-on and three discussion-based. These exercises task the participants with completing various competition relevant challenges. Each laboratory exercise includes the following sections: (1) a specification for any prerequisite technology needed, (2) learning objectives, (3) a mapping to relevant knowledge, skills, and abilities from the NIST NICE Cybersecurity Workforce Framework, (4) background necessary to complete the exercise, (5) the expected completion time, (6) configuration and setup steps needed which includes an initialization script where needed, (7) the challenges for the exercise, and (8) solutions to the challenges. The topics for the laboratory exercises in CYOTEE are directly motivated by common topics at cyber defense competitions. While the project is specifically targeted for competition preparation, it addresses core cybersecurity concepts which can be utilized outside of competition preparation or the academic environment.

Acknowledgements

I would like to acknowledge the following individuals and organizations for their respective roles in my accomplishments.

Daniel Conte de Leon, Ph.D.

Daniel has served as my mentor, coach, adviser, and major professor. Daniel has placed more confidence in my ability to be successful than I did in myself at times. He pushed me. He congratulated me. He has stood by me through my academic experience and has prepared me to leave academia and enter into the world of professional research. Daniel's drive, passion, and sense of humor are not overlooked, as those are qualities of his which I hope to carry on wherever I go and pass on to those whom I will have the opportunity to mentor, coach, advise, and teach. Thank you, Daniel.

Jim Alves-Foss, Ph.D. and Michael Haney, Ph.D.

Jim and Michael have offered me a great deal of technical knowledge related to cybersecurity. Both are brilliant individuals and excellent professors. Outside of the classroom, both have donated their time to my needs including course work, job hunting, interpersonal skill development, and various discussions on computer science topics. I could not have reached this point in my academic career without their support. Thank you, Jim and Michael for taking the time and energy to be on my committee.

Jessica Smith, Ph.D.

Jess has served as my boss, project sponsor, friend, and colleague. Jess has seen my technical ability grow from being my sponsor for Senior Design to being my mentor through my internships. Her ability to balance giving me the level of autonomy needed to grow but simultaneously provide an appropriate amount of guidance is almost uncanny, but I am thankful for it. She has constantly reminded me of all which I am capable of accomplishing: sometimes through words ("You got this!") and sometimes through actions ("I'm assigning you as task lead!"). Her professional guidance, technical prowess, sense of humor, and mentor ability are all traits which I hope to have in a boss throughout my career. Thank you, Jess.

Victor House

Victor has been the systems administrator at the University of Idaho longer than I have been at the university. Through those years, he has not lost a step. Victor is possibly the most reliable individual I know. From the late nights and early mornings configuring RADICL for a last minute tutorial to the amount of time he spends in the summer and winter ensuring RADICL is operational for the new semester, Victor does not say no. I have thoroughly enjoyed discussing computer science, video games, movies, and everything in between with Victor. Without his support and humor, I could not have accomplished all that I have. Thank you, Victor.

NSF CyberCorps Scholarship for Service Program

I would like to thank all the individuals involved with the SFS program at the University of Idaho as well as the national program. This program has afforded me opportunities which a college student can only dream of. From various trips to Washington D.C., to presenting at conferences, and funding my research, the scholarship has been invaluable in my academic venture.

Elizabeth Biancosino

I must thank my fiancée for her love, support, encouragement, and patience throughout this process. She has always helped me stay grounded, especially while I performed my master's degree research and wrote my thesis. She has reminded me that I cannot, in fact, stay up all night writing my thesis. She has sat through many boring computer science presentations of mine as a show of her support. In addition to being my biggest cheerleader, she is also my biggest role model. Her drive, passion, ambition, and ability to accomplish anything she puts her mind to is inspirational and motivational. Liz has been patient with me through long days in the lab and late nights working on research. She has supported me in every academic and professional decision I have made and without a doubt, I could not have reached this accomplishment without her by my side.

Quinn Wright-Mockler

I would like to thank my friend, colleague, and cubicle partner, Quinn, for his insight into what color scheme for highlighted text was most accessible for individuals with certain types of color blindness.

Friends and Family

This section would not be complete without thanking all my friends and family who have supported me through this process. To my friends who reminded me to take a break every now and then and just have fun. To my family who supported me and pushed me to reach all my academic goals.

Dedication

This thesis is dedicated to my late grandfather. A man who exuded brilliance and wisdom right up to his last day. He firmly believed that a formal education is the most powerful tool any individual can possess. He always encouraged me to achieve more than I thought I could. I recall many conversations with him in which he emphasized how important it was that I earned a master's degree. Today, I am pleased to have accomplished my goal, his goal, our goal. My only regret is that he is no longer with us to witness this accomplishment. Thank you for your support, belief, and encouragement.

For you, Dadu.

Table of Contents

List of Tables

List of Figures

List of Code Listings

List of Acronyms

AD	Active Directory
CA	Certificate Authority
CDC	Cyber Defense Competition
CTF	Capture the Flag
CYOTEE	CYbersecurity Oriented Training Environment and Exercises
DHCP	Dynamic Host Configuration Protocol
DMZ	Demilitarized Zone
DNS	Domain Name Service
GPO	Group Policy Object
GUI	Graphical User Interface
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
ICS	Industrial Control System
IP	Internet Protocol
IT	Information Technology
LAN	Local Area Network
NAT	Network Address Translation
NCCDC	National Collegiate Cyber Defense Compeition
NIST	National Institute of Standards and Technology
OS	Operating System

PC	Personal Computer
PEU	Pink Elephant Unicorn
PRCCDC	Pacific Rim Collegiate Cyber Defense Competition
SQL	Structured Query Language
SSH	Secure Shell
SSL	Secure Sockets Layer
VM	Virtual Machine
XSS	Cross Site Scripting

Glossary of Frequently Used Terms

CTF Capture the Flag competitions are a type of competition. Typically in a CTF, participants solve a challenge and are rewarded with a flag (a string of text). This flag is then submitted to earn points. Most CTFs have challenges in various categories with increasing level of difficulty.

Cryptography Cryptography is the process of encrypting and decrypting messages to ensure security. Encryption is the process of converting human comprehensible plain text to a non comprehensible cipher text.

CDC Cyber Defense Competitions are events in which blue teams compete against a red team. At these competitions, blue teams are given vulnerable environments and are tasked to secure and defend their environment against active attacks for the red team. The red team is comprised of cybersecurity professionals.

Cybersecurity Cybersecurity is the science of defending, protecting, and securing computing environments from threats in cyber space (the Internet).

Database A database is a structured set of information (data). In this context, databases are stored and maintained on a computer.

Directory A directory is a folder on a computer.

Domain A domain is a group of computers and users that are remotely managed based on a set of rules and policies.

Environment An environment in this context is the collection of the network, computers, and physical objects used in an organization.

Exploit An exploit, in this context, is an event or action which takes advantage of a vulnerability to cause damage to a machine or organization.

HTML Hypertext Markup Language (HTML) is a programming languages used to write web pages. HTML is used to provide the content for a web application.

Linux Linux is an open source operating system based on UNIX.

Machine A machine is any computer or device in an environment. This includes desktops, laptops, cell phones, etc.

MySQL MySQL is a database management tool. MySQL allows a user to interact with a database to read contents, insert data, remove data, and perform other actions.

Networking Networking is the action of connecting computers and devices to each other through devices such as routers and switches both physically and wirelessly.

Nginx Nginx, among other things, is an application used to serve web pages.

OS An operating system (OS) is the software that supports a computer's basic functions, such as scheduling tasks, executing applications, and controlling peripherals.

PHP PHP is a language used in web applications to enable communication between the web application and a corresponding database.

Query A query is an action performed on a database. Queries allow a user to insert, read, remove, and modify data in a database.

Risk Risk, in this context, is the possibility for a machine or organization to be exposed to danger.

Server A server is a computer which performs a specialized or central task such as hosting a web page or database.

Terminal A terminal is an application, commonly found in Linux and UNIX, which allows a user to enter commands sent directly to the operating system to perform an action.

Threat A threat is any event or action which presents the potential for damage to a machine or organization.

Ubuntu Ubuntu is an open source Linux-based operating system developed by Canonical.

UNIX UNIX is an early operating system designed by Bell Labs.

User The user is the human who is using a machine.

Virtualization Virtualization is the act or creating a virtual version of hardware, operating systems, or other computing devices.

VM A virtual machine is a machine which has been virtualized to perform the tasks of the virtualized environment.

VMware VMware is a virtual machine manager which can be used to create and run virtual machines.

Vulnerability A vulnerability, in this context, is an aspect of a machine or organization which can be exploited.

Windows Windows is an operating system developed by Microsoft.

Chapter 1: Introduction

1.1 Background

Cyber defense competitions (CDCs) provide students with a hands-on, real world opportunity to learn, practice, and perform the tasks which they will be expected to complete in the workplace. CDCs include red-blue exercises (attack-defend), capture the flag competitions, Jeopardy style competitions, level based training exercises, and environment configuration type activities. In a traditional CDC, commonly referred to as red-blue or attack-defense competitions, the blue team (team of students) defend an infrastructure against a red team (team of professionals) with the goal of compromising the student team's infrastructure. The blue team are challenged with hardening and securing an infrastructure comprised of various machines containing vulnerabilities. Simultaneously, the red team will attempt to discover the aforementioned vulnerabilities in the blue team's machines and perform exploits to compromise the infrastructure.

Another class of competitions is the "capture the flag" competition. At a capture the flag competition, participants complete various challenges including, but not limited to, decrypting messages, reverse engineering executables, discover a hidden message in an image, or hack a web page. By completing these challenges, participants are presented with a "flag", typically a string of text, which can then be entered for points.

Yet another class of competitions is the "Jeopardy style" competition. This type of competition is designed with multiple categories of challenge with increasing point values correlated to the difficulty of the challenge. Although not necessarily required, Jeopardy style competitions are frequently used in capture the flag competitions as the backbone for the challenge categories and flag point values.

Although not necessarily a competition, "level based training exercises" can be used to learn or refine skills. This type of training begins by tasking the participant to complete a basic level challenge with increasingly complex challenges as levels increase. This type of exercise is often more of a game, rather than a competition but can be used in a competitive environment.

"Environment configuration" type activities task the participant to perform various configu-

ration steps on a machine, network, or system. Tasks will vary based on the environment being configured. In a network configuration activity, challenges may include network segmentation and inbound/outbound firewall rule creation. In a domain configuration activity, on the other hand, challenges may include create user groups and organizational units as well implementing and deploying group policy objects.

Related projects include the SEED Labs, the NIST NICE Challenge, EDURange, Incident Response Training Scenarios, and Online Training Coursework. The SEED Labs are hands-on laboratory exercises which focus on exploiting common security vulnerabilities including SQL injection, cross site scripting, and buffer overflows. The NIST NICE Challenge is an online exercise which participants schedule a time to complete in the cloud and includes a set of challenges which map directly back to the NIST NICE Cybersecurity Workforce Framework. EDURange is a project in which participants complete various exercises on cloud servers with discussion questions following exercise completion. Incident Response Training Scenarios provide participants with the opportunity to discuss various responses to example scenarios of cyber incidents, as a real incident can be challenging to design and implement in the real world. Various organizations have created sample incidents with related discussion questions. Online Training Coursework can be completed to achieve an in-depth understanding of the topic being covered in the course. Various organizations provide online training coursework.

Cybersecurity standards have existed for decades with new standards being developed. Existing standards include the International Organization for Standardization (ISO) standards for cybersecurity, the Defense Federal Acquisition Regulation (DFARS), the Secure Controls Framework (SCF), and the National Institute of Standards and Technology's National Initiative for Cybersecurity Education (NIST NICE) Cybersecurity Workforce Framework. The NIST NICE Framework most relates to the objectives of this thesis as it provides a taxonomy of cybersecurity workforce roles and the knowledge, skills, and abilities needed to perform a work role effectively.

1.2 Problem

Cyber defense competitions are a good platform for learning, practicing, and refining vari-

ous cyber defense techniques. Cyber defense competitions enable students to have a real world experience in an enterprise information technology (IT) environment. This real world experience includes hardening vulnerable machines, maintaining uptime of critical services, providing technical support, documenting and reporting on incidents, and responding to public relations requests, all while under duress from red team attacks.

In order for students to get the most from their experience at a cyber defense competition, they should be adequately prepared in not only the basic concepts of cybersecurity, but also core skills needed at cyber defense competitions. The availability of current training material for cyber defense competitions is limited, especially when considering the subset of materials which are targeted for cyber defense competitions, freely available, and modifiable for extended use.

1.3 Proposed Solution and Contributions

1.3.1 Proposed Solution

CYbersecurity Oriented Training Environment and Exercises (CYOTEE), my contribution, is a project intended to fill the need for preparatory material targeted for cyber defense competitions. CYOTEE is a freely available project which will be available on a public GitHub repository. The virtual machines (see chapter 4) which are used in CYOTEE are modifiable and can be extended past the associated laboratory exercise (see chapters 5-13) for further preparation.

The specific contributions described in this thesis are nine laboratory exercises (six hands-on and three discussion-based) which task the participant with completing various competition relevant challenges.

The laboratory exercises address concepts which relate to common tasks required of a blue team at a CDC. The laboratory exercises presented in CYOTEE cover the following topics: the Linux terminal, Linux hardening, MySQL hardening and usage, web application security, active directory, customer service, organizational management, and incident response.

Each laboratory exercise includes the following items: (1) a specification for any prerequisite

Contribution #	Chapter in Thesis	Title
Contribution 1	Chapter 4	Configuration, Setup, and Environment Selection
Contribution 2	Chapter 5	Basics of the Linux Terminal
Contribution 3	Chapter 6	Linux Hardening
Contribution 4	Chapter 7	MySQL Usage & Hardening
Contribution 5	Chapter 8	Creating a Vulnerable Web Application
Contribution 6	Chapter 9	Web Application Hardening
Contribution 7	Chapter 10	Active Directory Usage & Hardening
Contribution 8	Chapter 11	Performing Customer Service at Competitions
Contribution 9	Chapter 12	Organizational Management Tasks Applied at Competitions
Contribution 10	Chapter 13	Introduction to Incident Management and Response

Table 1.1: *Contributions by Chapter and Title*

technology needed, (2) learning objectives, (3) a mapping to relevant knowledge, skills, and abilities from the NIST NICE Cybersecurity Workforce Framework, (4) background necessary to complete the exercise, (5) the expected completion time, (6) any configuration and setup steps needed which includes an initialization script where needed, (7) the challenges for the exercise, and (8) solutions to the challenges.

The topics for the laboratory exercises in CYOTEE are directly motivated by common topics at cyber defense competitions. CYOTEE was developed with the intent of supporting students in preparing effectively for cyber defense competitions. CYOTEE laboratory exercises and the associated initialization script are free, removing financial ability as a factor in ability to access preparatory material. After completing a CYOTEE laboratory exercise, a student or proctor can modify the virtual machine to add more vulnerabilities or using the virtual machines in a red-blue exercise.

1.3.2 Contributions

The contributions (laboratory exercises) are listed in table **1.1**:

Chapter 2: Background

CYOTEE was created based on motivation from various sources. One goal of CYOTEE was to provide a holistic approach to learning by combining reading, lecture, and hands-on learning. Because CYOTEE is targeted at CDC preparation, multiple CDCs were used as motivation for the design of CYOTEE as well as the topics covered. In order to validate the learning accomplished by CYOTEE, learning outcomes needed to map to a standard or framework which defines core cybersecurity roles and the necessary concepts to effectively perform in that role in the workforce. This chapter discusses the various motivation and background for CYOTEE.

2.1 A Holistic Approach to Learning

It has been suggested that hands-on learning is important to a holistic approach to learning [10]. While a "best way" to teach has not been discovered, the integration of hands-on learning to modern curricula has been shown to be effective [10]. In addition, reading is another important method of learning which should not be ignored [12]. CYOTEE support both ways of learning, hands-on and reading.

CYOTEE supports hands-on learning by tasking students to complete challenges in each laboratory exercise. These challenges require students to mitigate against vulnerabilities, configure machines, utilize applications, and discuss cybersecurity concepts. Students have the opportunity to learn about and practice various cyber defense skills through the laboratory exercises.

CYOTEE supports learning via reading by providing the students downloadable oratory exercises. The lab exercises include tasks along with an explanation of the vulnerability associated with the task. In addition to the tasks, each lab exercise includes with a guided walkthrough which outlines step by step how to complete each challenge in detail as well as provide rationale for the steps outlined. The guided walkthrough can be used either individually for self-guided study, or by a professor or proctor when teaching a course.

2.2 What is a Cyber Defense Competition?

Traditional cyber defense competitions (CDCs) are training exercises in which a team of attackers exploit vulnerabilities in the infrastructure of a team of defenders. The team of attackers are known as the red team while the defending team is known as the blue team; these training exercises have also been referred to as red-blue exercises. In a typical CDC, there may be multiple blue teams and multiple red teams.

It is standard at CDCs that the blue team receives an inherently vulnerable infrastructure. There is often a story line spun around the competition in which the previous information technology (IT) team was unable to defend the infrastructure so the blue team has been called in to harden the vulnerable systems and defend against attacks from the red team. Vulnerabilities may include weak or default passwords, malicious accounts, and back doors. It is the task of the blue team to identify as many vulnerabilities of the inherited infrastructure as possible and subsequently patch or harden them so as to keep the red team from exploiting the vulnerabilities.

The infrastructure at these competitions is typically made up of common services. Services are the applications which the blue teams are tasked with securing. Services include common applications such as database servers, web servers, domain controllers, firewalls, and workstation hardening. A database server contains information relating to the story line in the competition. For example, if the story line were that the IT team was working for a professional athletic team, the database may contain athlete statistics. A web server manages the website for the organization. If the athletic team example, the website may contain a schedule, ticket information, and athlete statistics which are drawn from the database. A domain controller can be used to manage multiple machines on a domain. In an enterprise environment, there are many computers, all which must comply with company security policy. A domain controller can be used to push the policy out across the entire domain to ensure that all machines are appropriately configured. Firewalls provide IT teams with a tool to systemically control what network traffic is allowed both inbound and outbound using various different filters including, but not limited to, Internet Protocol (IP) addresses and network protocols. Workstations are

the individual computers which employees work at on a daily basis. These machines, while not necessarily running any critical services, can contain sensitive information and have access to systems which run critical services. Ensuring that workstations are properly configured and secured is vital to a secure infrastructure.

These services are scored based on what is known as "up-time". At competitions a score bot checks to see if the service is up, running, and accessible, if yes, then the team receives up-time points for that service. If a service is down, it may be indicative of the system being compromised by the red team. Blue teams must then work to identify, detect, and remove the intruder from the impacted machine, at which time they will resume earning up-time points. In addition to keeping services up, blue teams must also perform business tasks to emulate the corporate side of IT. These tasks include answer phone calls, responding to emails, maintaining logs, reporting incidents, and responding to requests from corporate.

2.3 Cybersecurity Standards and Frameworks

Various organizations have set out to standardize and define the core concepts surrounding the field. In the early 2000s, the International Organization for Standardization (ISO) released the first revision of the ISO 27000 family of standards [106]. The 27000 family of ISO standards outlines standardization for information security management systems [104, 105]. Revisions have been made to the various standards within the family in addition to new standards being created [104, 105, 106].

Another set of standards in the field of cybersecurity is the Defense Federal Acquisition Regulation (DFARS) [107]. The DFARS set standards and regulation for Federal acquisitions across the U.S. Department of Defense [108]. The DFARS are updated and amended on a regular basis to keep up with modern cases, including up to the present date in November 2019.

In 2018, the Secure Controls Framework (SCF) was created [111]. The SCF aims to create a standardized taxonomy to help individuals in different cybersecurity related roles to speak a common language [111]. The SCF specifically aims to taxonomize the various types of obligations an organization may have. These obligations include statutory obligations, regulatory

obligations, contractual obligations, and industry-recognized leading practices [109, 110].

The National Institute for Standards and Technology (NIST) developed the National Initiative for Cybersecurity Education (NICE) Cybersecurity Workforce Framework [54, 55]. The NIST NICE Cybersecurity Workforce Framework is specifically targeted at defining and specifying the various knowledge, skills, and abilities an individual must possess to successfully integrate into the workforce in the respective work role. Because cyber defense competitions aim to provide students with a real world experience in information technology (IT) to prepare them for integration into the workforce, the NIST NICE Cybersecurity Workforce Framework is utilized in CYOTEE to map the learning outcome of each laboratory to the relevant knowledge, skills, and abilities.

The NIST NICE Cybersecurity Workforce Framework is comprised of Categories, Specialty Areas, and Work Roles. Each Work Role contains a set of knowledge, skills, and abilities which are necessary for effective completion of tasks in that role [54, 55].

The entire NIST NICE Cybersecurity Workforce Framework is detailed in NIST Special Publication 800-181 [54]. As the threat posed by cyberattacks grows, the need for a collective, knowledgeable, and well prepared cybersecurity workforce grows as well. The NIST NICE Cybersecurity Workforce Framework supports "a partnership between government, academia, and the private sector working to energize and promote a robust network and an ecosystem of cybersecurity education, training, and workforce development" [54].

By employing the NIST NICE Cybersecurity Workforce Framework across training exercises, a more uniform and consistent workforce is developed. The U.S. Department of Homeland Security's National Initiative for Cybersecurity Careers and Studies [56] and the SANS Institute [57], along with NIST, outline the Categories, Specialty Areas, and Work Roles in detail.

The NIST NICE Cybersecurity Workforce Framework contains seven Categories [54, 55, 57], namely:

1. Analyze
2. Collect and Operate
3. Investigate

4. Operate and Maintain
5. Oversee and Govern
6. Protect and Defend
7. Securely Provision

The three categories most relevant to CYOTEE are Protect and Defend, Operate and Maintain, and Securely Provision. These categories have the following specialty areas, respectively [54, 55, 57]:

Protect and Defend

1. Cyber Defense Analysis
2. Cyber Defense Infrastructure Support
3. Incident Response
4. Vulnerability Assessment and Management

Operate and Maintain

1. Data Administration
2. Knowledge Management
3. Customer Service and Technical Support
4. Network Services
5. Systems Administrator
6. Systems Analysis

Securely Provision

1. Risk Management
2. Software Development
3. Systems Architecture
4. Systems Development
5. Systems Requirements Planning
6. Technology R&D
7. Test and Evaluation

Diving deeper, the specific specialty areas (along with the specific Work Role ID) most relevant to CYOTEE include the following (see Appendix A for the NICE Specialty Area Descriptions associated with each specialty area below) [54, 55, 57]:

- Incident Response (CIR), PR-CIR-001
- Cybersecurity Defense Infrastructure Support (INF), PR-INF-001
- Data Administration (DTA), OM-DTA-001
- Customer Service and Technical Support (STS), OM-STS-001
- Software Development (DEV), SP-DEV-001, SP-DEV-002
- Technology R&D (TRD), SP-TRD-001
- Test and Evaluation (TST), SP-TST-001

Delving another level deeper into the NIST NICE Cybersecurity Workforce Framework, specific knowledge, skills, and abilities (KSAs) are associated with each Work Role. The following KSAs (along with their KSA ID) associated with their respective Specialty Area and Work Roles are most relevant to CYOTEE [54, 55, 57]:

- Incident Response (CIR), PR-CIR-001

  - K: Cybersecurity and Privacy Principles, K0004
  - K: Cyber Threats and Vulnerabilities, K0005
  - K: Application Security Threats and Vulnerabilities, K0070
  - S: Recognize Types of Vulnerabilities, S0078
- Cybersecurity Defense Infrastructure Support (INF), PR-INF-001
  - K: Basic System and OS Hardening Techniques, K0205
  - S: System, Network, and OS Hardening Techniques, S0121
  - S: Apply Cybersecurity and Privacy Principles to Organizational Requirements, S0367
  - A: Apply Cybersecurity and Privacy Principles to Organizational Requirements, A0123
- Data Administration (DTA), OM-DTA-001
  - K: Cybersecurity and Privacy Principles, K0004
  - K: Cyber Threats and Vulnerabilities, K0005
  - K: Data Administration, K0020
  - K: Database Management Systems, K0023
  - K: Query Languages, K0069
  - K: Database Access Application Programming, K0197
  - K: Database Theory, K0420
  - S: Conducting Queries, S0013
  - S: Generate Queries, S0037
  - A: Maintain Databases, A0176
- Customer Service and Technical Support (STS), OM-STS-001
  - K: Cybersecurity and Privacy Principles, K0004
  - K: Cyber Threats and Vulnerabilities, K0005

  - K: Electronic Devices, K0114
  - K: File Extensions, K0116
  - K: Industry Best Practices for Service Desk, K0237
  - K: Organization's Information Classification Program, K0287
  - K: Basic Operation of Computers, K0302
  - K: Documenting Reported Incidents, Problems, and Events, K0317
  - A: Accurately Define Incidents, Problems, and Events, A0025

- Software Development (DEV), SP-DEV-001
  - K: Cybersecurity and Privacy Principles, K0004
  - K: Cyber Threats and Vulnerabilities, K0005
  - K: Programming Language Structures and Logic, K0068
  - K: Application Security Threats and Vulnerabilities, K0070
  - K: Secure Coding Techniques, K0140
  - S: Designing Countermeasures to Identified Security Risks, S0022
  - S: Developing and Applying Security System Access Controls, S0031
  - S: Writing Code, S0060
  - A: Develop Secure Software, A0047

- Technology R&D (TRD), SP-TRD-001
  - K: Cybersecurity and Privacy Principles, K0004
  - K: Cyber Threats and Vulnerabilities, K0005
  - K: Application Vulnerabilities, K0006
  - K: Hacking Methodologies, K0310
  - S: Applying Secure Coding Techniques, S0172

- Test and Evaluation (TST), SP-TST-001

- K: Cybersecurity and Privacy Principles, K0004
- K: Cyber Threats and Vulnerabilities, K0005
- K: Test and Evaluation Processes, K0250
- S: Conducting Test Events, S0015
- S: Writing Code, S0060

The KSAs listed above are mapped to the individual laboratory exercises to highlight which KSAs a student should possess upon successful completion of the exercise.

2.4 Cyber Defense Competitions

In this section, the details of multiple CDCs will be discussed. The CDCs mentioned are those which directly motivated the topics addressed in CYOTEE. While motivation was drawn from all of the competitions discussed in this section, the bulk of motivation for the CYOTEE laboratory exercises comes from the National Collegiate Cyber Defense Competition (NCCDC) and the U.S. Department of Energy's CyberForce Competition. CYOTEE uses the skills required and tasks to be completed at competitions as guidance for the learning objectives in the laboratory exercises.

2.4.1 National Collegiate Cyber Defense Competition

The most nationally recognized CDC is the National Collegiate Cyber Defense Competition (NCCDC) [1]; NCCDC was recognized by the 111th Congress for its excellence [2]. In addition to government, NCCDC has received praise and support from industry and the students participating in the competition. Raytheon [42], a major U.S. defense contractor and industrial corporation, is the platinum sponsor of NCCDC has supported the competition for many years and has posted videos of students discussing the impact NCCDC has had for them [43, 45, 44, 46].

The teams which are selected to compete at NCCDC are determined through multiple regional competitions. The winners from each of the regional competitions are selected to partic-

ipate at NCCDC. NCCDC divides the United States into ten regions.

The University of Idaho competes in the Pacific Rim region and therefore attends the Pacific Rim Collegiate Cyber Defense Competition (PRCCDC). PRCCDC, held in the Seattle, WA area each spring, hosts 12 teams from Idaho, Oregon, and Washington. If more than 12 teams register for the competition, a prequalifying event is held to determine the top 12 teams.

Student teams are placed into a room in which there are typically eight machines running between eight and ten services. These services typically include Active Directory (AD), Domain Controller (DC), Web Server, Mail Server, Database Server, File Server, and a Human Machine Interface. In addition to these technical services, there is also a business representative for each team.

The technical services are initially all very vulnerable, and it is the job of the blue teams to identify and patch these vulnerabilities. While the blue teams are attempting to patch these vulnerabilities, a team of cybersecurity professionals, known as the Red team, attempt to exploit vulnerabilities which have gone unpatched to further wreak havoc on the services. In addition to defending the systems, business tasks known as injects are intermittently delivered to the blue teams.

The injects ask the teams to complete some sort of task such as adding or removing users to/from the domain, creating a list of users who have accessed a service in the last hour, or submitting an incident log. To make these injects easier to complete, blue teams are typically advised to keep logs of all changes they make on the host machines and keep a record of any security incidents which occur. In addition to managing these injects, the business representative is responsible for answering the team's telephone. Telephone calls range from a disgruntled employee to a wrong number asking about heating pizza rolls. Maintaining composure and being respectful while attempting to get back to work is a key task for the business representative.

Teams earn points by keeping their services online and accessible by the score-bot; these points are known as uptime points. Additionally, teams earn points by completing injects and responding in a respectful manner to phone calls and other requests. Teams can also earn points back from certain attacks if they are able to produce adequate logs of the events that took place, those who were involved, the services affected, and other relevant information [1].

2.4.2 U.S. Department of Energy CyberForce Competition

Another competition used as motivation and guidance for CYOTEE's learning objectives is the U.S. Department of Energy's (U.S. DOE) CyberForce Competition [4]. This competition has grown over the last two years from being held at a single location to being hosted nationwide at seven different U.S. DOE National Laboratories. The CyberForce competition, while similar to competitions such as PRCCDC and NCCDC, specifically targets cybersecurity of industrial control systems (ICS).

Critical infrastructure is an sector that has been targeted by cyber-attacks [3] recently and therefore, security surrounding critical infrastructure and industrial control systems is a major area of focus in the United States and worldwide at present [5, 6], making this theme relevant. Attacks on critical infrastructure incorporating malware such as Black Energy 3 and CrashOverride have shown that critical infrastructure is vulnerable and that the impact of a cyber attack on critical infrastructure can be significant [7].

The U.S. Department of Energy states, "Unfilled cybersecurity careers will reach over 1.5 million by 2019" [3]. The need to fill these positions with knowledgeable and adequately prepared individuals led to the CyberForce competition being created. The CyberForce competition aims to accomplish three goals: increase hands on education, increase awareness of critical infrastructure, and increase basic understanding of cybersecurity in a real-world scenario [3].

The CyberForce competition is unique in that it incorporates realistic ICS components into the competition. The competition has physical devices (ICS components) on the table react to the state of their network. For example, at the April 2018 competition, a light shone inside of a model Lego building, placed on the blue team's table, when the network was not compromised; upon becoming compromised, the light was shut off or set to flicker in various ways.

2.4.3 CyberPatriot

While PRCCDC/NCCDC and the CyberForce competitions are targeted at college level students, the U.S. CyberPatriot [8] is a program designed for K-12 students. The U.S. Cy-

berPatriot program was created by the Air Force Association in an effort to encourage young students to pursue careers in cybersecurity and other STEM fields. The CyberPatriot program is comprised of multiple different programs. The central program is the National Youth Cyber Defense Competition (NYCDC). Rather than have an attack-defend environment like the aforementioned competitions, students are given vulnerable operating systems as virtual images. The goal of this competition is to find as many of the vulnerabilities as possible and then harden the machines to patch the vulnerabilities. The student teams compete within their state and region for an opportunity to be invited to the National Finals Competition [8].

Aside from the NYCDC, the CyberPatriot program includes CyberCamps, the Elementary School Cyber Education Initiative (ESCEI), and CyberGenerations. The CyberCamps are held during the summer and teach students about cybersecurity and how they can apply it to their daily lives. The ESCEI brings awareness of cybersecurity principles for K-6 students via interactive learning modules. The CyberGenerations program is targeted at senior citizens to teach basic cyber hygiene and bring awareness to modern cyber threats which may impact them such as password management, social engineering, and phishing [8].

2.4.4 Pink Elephant Unicorn

Pink Elephant Unicorn (PEU) [15] is a friendly capture the flag cybersecurity competition hosted by Pacific Northwest National Laboratory. PEU uses the Facebook CTF [16] hosting platform with a modified map. PEU consists of trivia questions, challenges, and king of the hill. The trivia questions are meant to be able to be answered by browsing the internet. The trivia concepts map to basic cybersecurity concepts such as the CIA (Confidentiality, Integrity, and Availability) triad [17], the NIST Nice Framework [54], phishing, etc. The challenges require participants to perform some action such as decrypting a message (cryptography), finding hidden information in a photo (steganography), identifying network devices and drawing a network diagram (networking), or writing a script to accomplish a task (programming). The king of the hill portion of the competition has the participants exploiting vulnerable machines and claiming control of them. While in control, the participants should attempt to harden the machines

such that other participants cannot break in as well. If another team does break in and claim control of the machine, that team in control will begin to receive points for control of the "base" (terminology based on the Facebook CTF platform).

Because PEU is a largely an educational event, there are also tutorials/seminars which cover a range of cybersecurity topics such as cryptography, networking, cyberphysical security, reverse engineering, programming, etc. Participants can earn points by attending these tutorials. PEU has been running for five years, being hosted twice a year, once in Seattle, WA and once in the Richland, WA.

2.4.5 PicoCTF

Developed by Carnegie Mellon University, picoCTF [18] is an online cybersecurity competition which is held annually. Carnegie Mellon University. The development team states that "picoCTF is a free computer security game targeted at middle and high school students" [19]. The competition is Jeopardy style having multiple categories, each with varying difficulty levels. The categories include reverse engineering, scripting, cryptography, etc. The entire game is also spun around a unique story line meant to engage the participant. Although the competition is only live for approximately two weeks each year, the game is made available to be played year round. During the competition, participants can compete as individuals or teams, with prizes for the top teams. The competition also offers bug bounties to participants who are able to identify security bugs in the picoCTF framework [18].

2.4.6 Over the Wire

OverTheWire is an online practice ground for cybersecurity skills. The exercises range from simple basics of Linux up to complex ethical hacking exercises. The platform is available online at all times, and requires no registration. Students will need a computer from which they can SSH onto the OverTheWire servers. On Linux machines, SSH should come as a built-in service, but if not can be installed. On Windows machines, users can install an application called PuTTY. OverTheWire includes various "wargames", each with levels which the player

progresses through with increasing difficulty [20].

Chapter 3: Related Work

There are many projects which are related to CYOTEE. Related projects share certain qualities with CYOTEE including format of contributions (laboratory exercises), objectives of project (improve cybersecurity education and training material), topics (common topics seen at CDC). The following projects are related to the work done in CYOTEE.

3.1 SEED Labs

The SEED labs are a series of lab exercises in cybersecurity designed to increase the number of hands-on experiences in cybersecurity education [9]. The SEED labs are deployed as Ubuntu images which can be downloaded from the internet. There are over 30 labs which cover various cybersecurity topics including SQL Injection, Cross-site Scripting, and Buffer Overflows. In these labs, students exploit vulnerabilities in the applications being run on virtual image they downloaded to better understand the vulnerability and how to exploit them. The SEED labs include detailed summaries of the vulnerabilities and explain why a specific exploit works on the vulnerability present [9].

3.2 NIST NICE Challenge

The NICE Challenge Project is a set of challenges which are used across industry and academia. These challenges are based on the NIST NICE Framework [86] and include a variety of challenge types. The challenges in the NICE Challenge are narrative-driven scenarios, emulate full-scale business environments, and include technical objectives and written deliverables [86]. At present 375 educational institutes have used the NICE Challenge, with over 600 educational faculty signed up to deploy the NICE Challenge [86]. The NICE Challenge consists of virtualized business environments to provide a realistic experience. The NICE Challenge is used in various ways, including capstone experiences, laboratory exercises, competition preparation, and free play [86]. The NICE Challenge maps directly to the NIST NICE Framework including 101 unique challenges, addressing 12 Work Roles, and 261 KSAs [86]. The NICE Challenge Project is typically discussed in he NIST NICE quarterly newsletter. The summer 2019

newsletter discussed an increase in unique challenges as well a major redesign of the project's web portal [87]. The spring 2019 newsletter highlights an increase in the project's user base, marking the most rapid growth in K-12 sector [88].

3.3 Incident Response Training Scenarios

Cyber incidents can be difficult to combat, especially because they can catch an organization off guard. By running preparatory exercises, an organization can strengthen its incident response process. One problem that organizations can run into is knowing how to practice incident response when real incidents can be costly to emulate. For this reason, discussion based response using scenarios can be effective. Several resources are available for sample cyber incident scenarios.

One paper from the Center for Internet Security [97] includes six scenarios which can be used to train a team. This exercise is performed as a tabletop exercise and ranges in the type of incident. The exercises also include discussion questions.

In an article from Delta Risk [98], Ewing presents cyber incident scenarios which can be utilized. Ewing's incident topics include phishing emails, malicious attachments, suspicious requests, and unauthorized devices on a network. Ewings material provides topics, but not sample incidents. The facilitator or moderator will have to create a compelling scenario to accompany the topics.

A more comprehensive document, MITRE's Cyber Exercise Playbook [99] walks through the many stages of incident management. The document includes background on the types of exercises, threats, and planning cycle. The document includes scenario based exercises with takeaways and lessons learned from the exercises. The exercises include a short description of the incident, assumptions made, notes if applicable, and expected actions.

3.4 Online Training Coursework

CYOTEE is intended to be used as training material for cyber defense competitions, but is not a competition itself. Other training opportunities are available, including SANS and Udemy

courses. These courses are for-fee, ranging from a couple hundred dollars to upwards of several thousands of dollars.

SANS offers three courses which are particularly relevant to CYOTEE, all available in three options: OnDemand, SelfStudy, and Private Training. As of September 2019, the cost of each course ranges from $6,000 to $7,000.

- **SEC503: Intrusion Detection In-Depth** [100]

  This course covers topics including analyzing and detecting incidents.

- **FOR508: Advanced Incident Response, Threat Hunting, and Digital Forensics** [101]

  This course covers topics including incident response, adversary detection, and timeline analysis.

- **MGT512: Security Leadership Essentials For Managers** [102]

  This course covers topics including building a program, leading initiatives, and protecting data.

The SANS courses have a high price point, but are used worldwide and are an excellent resource for those with ability to purchase the courses.

A less costly option is to use Udemy courses. Udemy has multiple courses per topic for the various topics relevant to CYOTEE including Active Directory, MySQL, Web Applications, and Linux. The Udemy courses range from $20 to $200. Udemy has over 100,000 online courses and offer the ability to access courses for life after purchase, allowing students to complete courses at their own pace [103].

Chapter 4: Configuration, Setup, and Environment Selection

In this chapter, I discuss the initialization scripts associated with the laboratory exercises, the rationale for technology selection including the virtual environment, services to focus on, and operating systems utilized for each laboratory exercise.

4.1 Initialization Scripts for All Laboratory Exercises

Initialization scripts are utilized in this project to allow students to use an unmodified, referred to as "vanilla", ISO rather than needing to download a preconfigured ISO. While preconfigured virtual machine ISOs will be available for download, if students would rather use their own ISOs, this section includes the initialization scripts which can be run on a virtual machine running the operating system required for the respective laboratory exercise. The scripts perform basic configuration tasks including creating files, modifying configuration files, creating users, populating application data, and cloning the CYOTEE GitHub repository. These scripts will require that your virtual machines have access to the Internet in order to download certain packages, applications, and other dependencies. Each of the following scripts are titled with the laboratory exercise for which they correspond to. A copy of each script is provided in the respective laboratory exercise as well. The scripts are shown in listings **4.1 - 4.5**. The title of the laboratory exercise associated with each initialization script is shown in table **4.1**.

Listing 4.1	Basics of the Linux Terminal
Listing 4.2	Linux Hardening
Listing 4.3	MySQL Usage & Hardening
Listing 4.4	Creating a Vulnerable Web Application
Listing 4.5	Web Application Hardening

Table 4.1: *Exercise Titles and Respective Initialization Script*

Basics of the Linux Terminal Laboratory Exercise, Chapter 5

Listing 4.1: *Basics of the Linux Terminal Laboratory Exercise Initialization Script*

```
#!/bin/bash

#flag1
echo "flag:\$encryptedpasswordsarekeptin/etc/shadow" >> /etc/shadow
echo "aOmpy3tFOB96wIwKAgX7GImPUuD1mLPADWXeXhZF2Hk2j" >> /etc/shadow
echo "thYYBSkDkE6ZzvkkBmWHiBoYjiDnRR3t95eEl6A1xUUUH" >> /etc/shadow
echo "wIttm0hwZxWfUbltKXU5JiqIS6rBGvk4MjEWpKmtmq8DC" >> /etc/shadow
echo "SfSPbJrr6Ny5oM9tArAF7wLQ876liujtnDONXE237iZeX" >> /etc/shadow
echo "Fi4xqO3PtdCL2rdYR4E8JTvBVL46pxqx4d23u7004NwB6" >> /etc/shadow

#flag2
echo "thingsarenotalwaysastheyappear" > /bin/weirdfile

#flag3
echo "system" > /sbin/tilapia
echo "binaries" > /sbin/salmon
echo "aren't" > /sbin/bass
echo "fishy" > /sbin/hering

#flag6
echo "itsalwaysgoodtokeeparecordofwhatversionswereinstalled" >> /var/backups/apt.
    extended_states.0

#flag8
echo "therearenogamesonthismachine!" > /usr/games/.spaceinvaders

#flag9
echo "ufwissuitedforhostbasedfirewalls" > /lib/ufw/flaaaag

#flag10
echo "ysiyndwnjceaxmdowgeyfapkfc" > /media/flag
```

Linux Hardening Laboratory Exercise, Chapter 6

Listing 4.2: *Linux Hardening Laboratory Exercise Initialization Script*

```
#!/bin/bash

#overhead
sudo dpkg --configure -a
sudo apt-get install git -y
sudo apt-get install openssh-server openssh-client -y
sudo service ssh start
sudo rm -r CYOTEE
sudo git clone https://github.com/CenterForSecureAndDependableSystems/CYOTEE.git

#create users
useradd redteam
useradd guest

#assign passwords to the users
sudo echo -e "redteam\nredteam" | passwd redteam
sudo echo -e "guest\nguest" | passwd guest

#disable auto-updates
sudo rm /etc/apt/apt.conf.d/20-auto-upgrades
sudo rm /etc/apt/apt.conf.d/20auto-upgrades
sudo cp CYOTEE/CYOTEE_Code_Linux/20-auto-upgrades /etc/apt/apt.conf.d/20-auto-
    upgrades

#add a couple of cron jobs

sudo crontab -u thesis -l | { cat; echo "* * * * * touch ~/Desktop/sensitivefile"
    ; } | crontab - -u thesis
sudo crontab -u thesis -l | { cat; echo "*/2 * * * * rm ~/Desktop/sensitivefile";
     } | crontab - -u thesis
```

MySQL Basics & Hardening, Chapter 7

Listing 4.3: *MySQL Basics & Hardening Laboratory Exercise Initialization Script*

```
#!/bin/bash

sudo dpkg --configure -a

#install git
sudo apt-get install git -y;

#remove existing repo and clone git repo
sudo rm -r CYOTEE
sudo git clone https://github.com/CenterForSecureAndDependableSystems/CYOTEE.git

#install MySQL
sudo apt-get install -y mysql-server;

#create the MySQL users
sudo mysql -u root -e "CREATE USER 'randomuser'@'localhost' IDENTIFIED BY '
    password'";
sudo mysql -u root -e "CREATE USER 'redteamer'@'localhost' IDENTIFIED BY 'redteam
    '";
sudo mysql -u root -e "CREATE USER 'haxxor'@'localhost' IDENTIFIED BY 'haxxor'";
sudo mysql -u root -e "CREATE USER 'testuser'@'localhost' IDENTIFIED BY 'test'";

#create the unnecessary database
sudo mysql -u root -e "CREATE DATABASE dontlook";

#remove the vulnerable database if one already exists
sudo mysql -u root -e "DROP DATABASE vulndb";

#create the vulnerable database
sudo mysql -u root -e "CREATE DATABASE vulndb";

#grant all privileges to users
sudo mysql -u root -e "GRANT ALL PRIVILEGES ON vulndb.* TO 'root'@'localhost'";
sudo mysql -u root -e "GRANT ALL PRIVILEGES ON vulndb.* TO 'testuser'@'localhost'
    ";

#import vulndb sql file
sudo mysql -u root vulndb < CYOTEE/CYOTEE_Code/SQL/vulndb.sql;
```

Creating a Vulnerable Web Application, Chapter 8

Listing 4.4: *Creating a Vulnerable Web Application Initialization Script*

```
#!/bin/bash

sudo dpkg --configure -a


#install nginx
sudo apt-get install nginx -y

#install php
sudo apt-get install php-fpm php-mysql -y

#install MySQL
sudo apt-get install mysql-server -y
```

Web Application Hardening, Chapter 9

Listing 4.5: *Web Application Hardening Laboratory Exercise Initialization Script*

```
#!/bin/bash

sudo dpkg --configure -a


#grab the github repo
sudo apt-get install git
sudo rm -r CYOTEE
sudo git clone https://github.com/CenterForSecureAndDependableSystems/CYOTEE.git

#install nginx
sudo apt-get install nginx -y

#install php
sudo apt-get install php-fpm php-mysql -y

#install MySQL
sudo apt-get install mysql-server -y

sudo cp CYOTEE/CYOTEE_Code/VulnerableCode/* /var/www/html/

sudo mv /var/www/html/default /etc/nginx/sites-available/default

sudo mysql -u root -e "DROP DATABASE test";
sudo mysql -u root -e "CREATE DATABASE test";

sudo mysql -u root -e "CREATE USER 'newuser'@'localhost' IDENTIFIED BY 'newpass'"
    ;
sudo mysql -u root -e "GRANT ALL PRIVILEGES ON test.* TO 'newuser'@'localhost'";

sudo mysql -u root test < CYOTEE/CYOTEE_Code/SQL/test.sql

sudo service nginx restart
```

4.2 Rationale for Technology Selections

CYOTEE is a heavily technology-based project. In determining which technology would be utilized in CYOTEE, various factors were considered. In this section, the technology which were selected for use in CYOTEE are discussed.

4.2.1 Virtual Environment Selection

During the development of CYOTEE, products related to personal virtualization were needed; VMware [24] and VirtualBox [48] were considered. Both products have a free version, support many guest operating systems, and had the capabilities required by CYOTEE. Ultimately, the decision was made to use VMware due to familiarity with the product within the CYOTEE team and the ability to scale the project up using VMware's paid product, VMware Workstation Pro [24].

VMware, Inc., a technology company owned by Dell Technologies [47], offers many products related to computing. VMware Workstation can be used to provide students a sandbox type environment where they can tinker and test their virtual machines.

In this project, the goal is to create numerous ISO images which can then be spun up as new virtual machines. To configure the ISOs as necessary, a virtual machine needs to be configured appropriately. To manage these virtual machines, a virtual machine manager is needed. A virtual machine manager, more commonly referred to as a hypervisor within the virtualization community, is software which can run existing virtual machines or be used to create new virtual machines. VMware Workstation is a hypervisor which runs on both Windows and Linux [24, 21]. Although VMware was used during development of CYOTEE, the exercises can be run using other hypervisors which include, but are not limited to, VirtualBox and KVM.

4.2.2 Service Selection

"Services", as they are commonly referred to at CDCs, are the applications or protocols which need to be managed by the students at cyber defense competitions. These services commonly

include mail server, file server, domain name service (DNS), and many more. The services chosen to be included in CYOTEE are ones which are commonly seen at competitions such as the National Collegiate Cyber Defense Competition [30, 29] and the U.S. Department of Energy's CyberForce Competition [4]. The services which were selected to be focused on in CYOTEE are based on the following topics:

- Linux workstation navigation
- Linux workstation hardening
- MySQL server management
- Active directory management
- Web server management
- Web application development

Services were selected based on how often they appeared in competitions and how realistically they could be emulated in a laboratory exercise.

4.2.3 Operating System Selection

Operating systems are specialized software which control the most basic functions of a computer including task scheduling, application execution, and peripheral device management, among others. Operating systems typically fall into one of two categories, "Desktop" or "Server" which each having its respective strengths and weaknesses when running certain applications. Desktop based operating systems include a graphical user interface (GUI) while server based operating systems are console (command line) only. Applications which may require graphical visualization are more appropriately run on a desktop based operating system while applications which only require a command line interface can be run on a server based operating system. In this section, the operating systems which were selected for the respective laboratory exercise(s) are discussed.

4.2.3.1 Ubuntu Desktop for Linux Hardening and Linux Terminal

Ubuntu [25] is an open source distribution of the Linux operating system (OS). Produced by Canonical, Ubuntu is one of the most popular flavors of Linux and is rated the best Linux distribution for powerful PC and laptops by Fossbytes [26]. Additionally, according to an infographic [49], Ubuntu had over 20 million launches in 2015, Ubuntu is found in smart phones, tablets, and even vehicles, and Ubuntu is employed by large corporations including Walmart, Netflix, Bloomberg and Dropbox [49].

Canonical has released two versions of the OS each year for the last few years, once in April and again in October. Ubuntu is an easy to download and install Linux distribution available from Canonical's website. To install Ubuntu in VMware, one can either follow the graphical installation process or allow VMware to perform an easy install [27], in which the user specifies the account username and password and allows the OS to be installed on its own.

A Linux desktop environment is a common machine provided to the students at CDCs. This machine may be used as a file server, web server, or general-purpose computer in competition. For CYOTEE, the Linux desktop virtual machine is running Ubuntu 16.04.6, an older version of Ubuntu released in 2016. Competitions regularly use older versions of operating systems as hardening older versions can be more challenging because patching may no longer be supported.

CYOTEE's Basics of the Linux Terminal laboratory exercise (chapter 5) is intended to familiarize the student with the Linux filesystem. In order to minimize the varying operating systems being used for the project, Ubuntu Desktop was appropriate. Desktop was used in liue of Ubuntu Server as the student can traverse the file system using the file explorer as well as the terminal.

The Ubuntu Desktop machine is used for CYOTEE's Linux Hardening laboratory exercise (chapter 6) as well as the Basics of the Linux Terminal laboratory exercise (chapter 5).

4.2.3.2 Ubuntu Server for MySQL

Every CDC includes a database, typically a MySQL [38] database; alternative database software include Postgres [50] and MariaDB [51]. Due to the prevalence of a MySQL database at competitions, MySQL was selected for the database portions of the CYOTEE lab exercises (chapters 5-13). Databases store information which can later be queried and are commonly coupled with a webpage to hold the information the webpage uses such as user information or account information. MySQL (My Structured Query Language) is a popular database management application which allows a user to create databases, tables, and entries, as well as modify and remove them. In addition, permissions can be granted to individual database users to ensure that security is maintained. MySQL uses a specific syntax for its queries and as a result, there is a moderately steep learning curve. Despite a steep learning curve, the application is still one of the most commonly used open source databases [28].

For virtualizing the database, Ubuntu Server 16.04.6, a variant of Linux, was selected. Although structured query language (SQL) databases can be managed via the use of software with a graphical user interfaces, one should be comfortable and familiar with managing a database from a terminal. Because Ubuntu Server has no graphical user interface, rather is strictly console based, it was decided to be appropriate for virtualizing the MySQL laboratory exercise.

4.2.3.3 Windows Server for Active Directory

Many CDCs have the blue teams emulate the IT department in an enterprise environment. A mass management software like Active Directory is a commonplace in enterprise environments. Active Directory may be used to configure machines and/or users on a domain under the same policy, known as group policy. Additionally, users can be created, placed into groups, and assigned different permissions by group. For this reason, it is crucial that the student is familiar with using Active Directory. Windows Server 2008 was selected for virtualizing Active Directory.

The Windows Server machine is used in the Configuring a Windows Server laboratory exercise.

4.2.3.4 Ubuntu Server for Web Server and Web Application

Any enterprise environment will have a public facing website. The integrity and availability of the external website are the most critical components of website related public relations. Availability is critical as a client, business partner, or potential employee must be able to visit your website to gather the necessary information they need. Integrity is critical as you want to ensure that the information on your webpage is correct. In addition to having the proper information, ensuring that your website has not been defaced is vital. Webpage management can be done through the use of a graphical user interface or via the console (terminal). The walkthrough for this laboratory exercise was written for the student to perform the exercise via a console. For this reason, using some version of Ubuntu Server is appropriate.

Ubuntu Server 16.04.6 is used for virtualizing the web server in the Creating a Web Application and Web Application Hardening laboratory exercises.

4.2.4 Networking Services

CYOTEE provides students with the opportunity to practice securing individual virtual machines (VMs) and services in the laboratory exercises. In addition to this, an overlying goal of CYOTEE is to provide students with the ability to host their own CDC. In order to create the environment for a CDC, the student will need to create vulnerable virtual machines (can use the virtual machines used in the CYOTEE laboratory exercises) and statically assign IP addresses. At CDCs, all of the machines and services provided to the blue teams are placed on the same subnet; to emulate CDCs, the VMs should be placed on the same subnet. Ensure that machines are able to communicate with each other by attempting to ping each of the other machines on the network.

4.2.5 Types of Networking VMs in VMware

Virtual machines can be networked together in numerous ways. Three common methods of networking virtual machines together are network address translation (NAT), host-only, and same subnet static IP assignment.

When networking using NAT, the virtual machine simply shares the IP address of the host machine, which is the physical machine on which the virtual machine is running. This is just like connecting the virtual machine to the internet.

When networking using host-only, a private network is created within that host. Other virtual machines on the same private will be able to communicate on this private network; however, unless a router is configured for the purpose, access to any other networks will not be available. This adds another layer of safety to the exercise environment.

When using same subnet static IP assignment, the user must provide an IP address, a subnet mask, and a primary gateway. VM to VM communication is still limited to those which are running on the same host machine and additionally, only machines on the same subnet will be able to communicate with each other [23].

Chapter 5: Basics of the Linux Terminal

In this laboratory exercise, the student will be introduced to the basic usage of the Linux Terminal.

5.1 Laboratory Exercise

Basics of the Linux Terminal

5.1.1 Specifications

This exercise requires that students have access to a Linux machine of some sort. Ubuntu Linux was used during development for this exercise. Because the exercise focuses on the student using command line techniques, this exercise will be performed on an installation of Ubuntu Server 16.04.6. The machine will be preconfigured with many "flags" placed in various locations which the students will attempt to find.

5.1.2 Learning Objectives

- Basic Linux/Unix terminal commands
- Understanding of the Linux file system
- Understanding of key Linux directories

5.1.3 Mapping to NIST Nice Framework

This laboratory exercise is intended to familiarize the student with Linux, the Linux terminal, and the Linux filesystem. This exercise should be completed prior to attempting the other exercises. This laboratory exercise maps to the following KSAs from the NIST NICE Framework:

- Electronic Devices (K0114)

5.1.4 Necessary Background and Expected Completion Time

This laboratory exercise can be completed by students with varying background and experience. The following categories should help identify approximately how much time (in minutes) will be necessary to complete the laboratory exercise, for a student meeting the criteria for the respective experience level.

- Beginner: A student in this category has little to no experience with Linux, the Linux terminal, and the Linux filesystem.
- Intermediate: A student in this category has experience with Linux and the Linux terminal but is unfamiliar with the Linux filesystem.
- Advanced: A student in this category has experience with Linux and the Linux terminal as well as familiarity with the Linux filesystem.

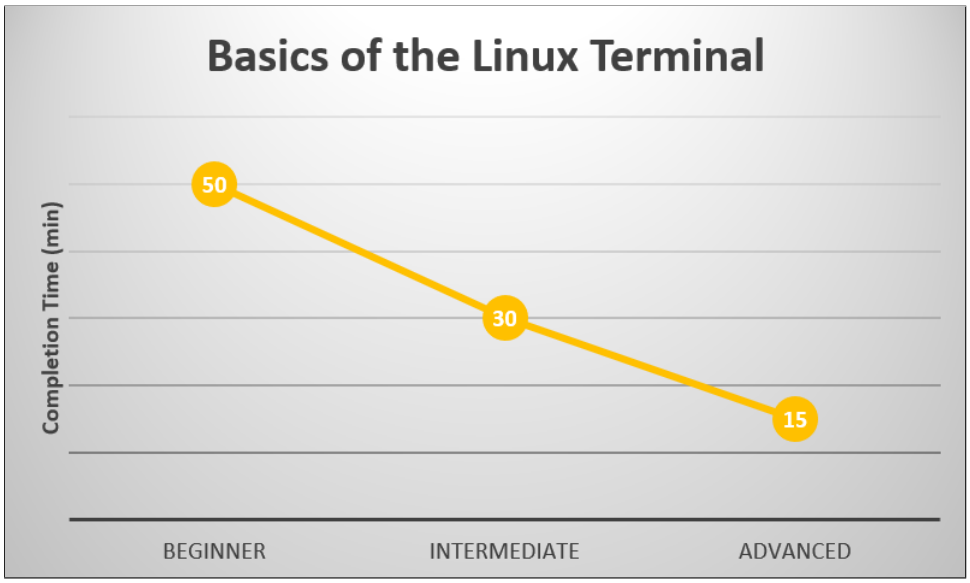

Figure 5.1: *Basics of the Linux Terminal Laboratory Exercise Expected Completion Time (min)*

5.1.5 Configuration and Setup

The machine used in this laboratory exercise is an installation of Ubuntu Server 16.04.6. It is pre-loaded with hidden flags across the filesystem (as performed by the initialization script). This machine is configured using the intialization script, listing **5.1**

Listing 5.1: *lt-initializationscript.sh*

```
#!/bin/bash

#flag1
echo "flag:\$encryptedpasswordsarekeptin/etc/shadow" >> /etc/shadow
echo "aOmpy3tFOB96wIwKAgX7GImPUuD1mLPADWXeXhZF2Hk2j" >> /etc/shadow
echo "thYYBSkDkE6ZzvkkBmWHiBoYjiDnRR3t95eEl6A1xUUUH" >> /etc/shadow
echo "wIttm0hwZxWfUbltKXU5JiqIS6rBGvk4MjEWpKmtmq8DC" >> /etc/shadow
echo "SfSPbJrr6Ny5oM9tArAF7wLQ876liujtnDONXE237iZeX" >> /etc/shadow
echo "Fi4xqO3PtdCL2rdYR4E8JTvBVL46pxqx4d23u7004NwB6" >> /etc/shadow

#flag2
echo "thingsarenotalwaysastheyappear" > /bin/weirdfile

#flag3
echo "system" > /sbin/tilapia
echo "binaries" > /sbin/salmon
echo "aren't" > /sbin/bass
echo "fishy" > /sbin/hering

#flag6
echo "itsalwaysgoodtokeeparecordofwhatversionswereinstalled" >> /var/backups/apt.
    extended_states.0

#flag8
echo "therearenogamesonthismachine!" > /usr/games/.spaceinvaders

#flag9
echo "ufwissuitedforhostbasedfirewalls" > /lib/ufw/flaaaag

#flag10
echo "ysiyndwnjceaxmdowgeyfapkfc" > /media/flag
```

5.1.6 Vulnerability List

This laboratory exercise does not contain any vulnerabilities. It's intended purpose is to familiarize the student with the Linux filesystem and navigating through Linux. The skills in this exercise will be necessary to perform the other exercises.

5.1.7 Challenges

Find as many of the flags hidden on the associated Linux machine as possible. The tasks do not have to done in order. Use the hints provided for finding each flag.

1. Flag 1 can be found where the configuration files are typically kept. Look where the passwords are cryptic.

2. Flag 2 can be found where user binaries are typically kept. Something is out of the ordinary.

3. Flag 3 can be found where system binaries are typically kept. Something seems fishy here.

4. Flag 4 can be found where device files are typically kept. Pages are typically 4K in size, but not always, where are larger pages handled?

5. Flag 5 can be found where process information is typically kept. The flag is the number of blocks in `sda2`. The number found will likely differ from the answer in the solution guide because the number of blocks in a partition varies from machine to machine.

6. Flag 6 can be found where variable files are typically kept. It is important to have a way to recall how things were earlier.

7. Flag 7 can be found where temporary files typically kept. You may have to ask about this one, the flag may be gone.

8. Flag 8 can be found where user programs are typically kept. Where might you find space invaders, maybe the flag is playing a popular childrens game (hide and seek) with you?

9. Flag 9 can be found where the system libraries are typically kept. Keep intruders out with a firewall.

10. Flag 10 can be found where removable devices are typically kept. You may have to decrypt the flag, what was the name of the cipher used? Vinegar or something like that. The key for the cipher is the name of the file which contains the ciphertext.

5.2 Solutions and Guided Walkthrough

5.2.1 Solutions

1. Flag 1 can be found in the `/etc/shadow` file. The flag is: **encryptedpasswordsarekept-in/etc/shadow**. Navigate to a directory by using the `cd` command. Ex: `$ cd /etc/`. To view the contents of a file, use the `cat` command. Ex: `$ cat /etc/shadow`.

2. Flag 2 can be found in `/bin/weirdfile`. The flag is: **thingsarenotalwaysastheyappear**. In order to list all the files in the current directory (folder), use the `ls` command. Ex: `$ ls /bin/. .`

3. Flag 3 can be found in the `/sbin` directory. There are four files named **tilapia, salmon, bass, and hering** each containing a part of the flag. The flag is: **systembinariesaren'tfishy**. Commands in Linux can be run in sequence by separating each command using the semicolon (;) character. Ex: `$ cat tilapia ; cat salmon ; cat bass ; cat hering`.

4. Flag 4 can be found in the `/dev` directory. The flag is: **hugepages**.

5. Flag 5 can be found in the `/proc` directory. In the file `partitions`, you will find a partition sda2 with a number of blocks. The flag is whatever the number of blocks in sda2 is. **Note: Your machine may not have an sda2 partition.**

6. Flag 6 can be found in the `/var/backups` directory. In the file `apt.extended_states.0`, there are three lines commented out which contain the flag. The flag is: **itsalwaysgood-tokeeparecordofwhatversionswereinstalled**

7. Flag 7 can be found in the `/tmp` directory. Well, it would be if not for the fact that the /tmp directory is flushed regularly (hence the name). For this challenge, give the student the flag when they have identified why there isn not a flag there. The flag for this task will be: **tempfilesareremoved**.

8. Flag 8 can be found in the `/usr/games/` directory. In this directory, there is a hidden directory titled `.spaceinvaders`, inside is a file containing the flag. The flag is: **therearenogamesonthismachine!**. **Note: Your machine may have other games installed in this folder. If so, laugh about the inaccuracy of the flag**. Hidden files can be difficult to find. Using the normal `ls` command does not display these files. Using the `-a` option, `ls` will also display hidden files. Ex: `$ ls -a /usr/games/`.

9. Flag 9 can be found in the `/lib/ufw` directory. There is a file in this folder containing the flag. The flag is: **ufwissuitedforhostbasedfirewalls**.

10. Flag 10 can be found in the `/media` directory. Here there is a file containing a cipher text. The key used to decrypt the cipher text in a vigenere cipher is in the file name. The flag is: **thisiswhereusbdirvesappear**.

5.2.2 Guided Walkthrough

In order to complete the challenges in this laboratory exercise, see the steps in this guided walkthrough.

Most Linux-based OSs have the option for a terminal to be opened using the **Ctrl + T** hotkey. Additionally, if the OS has a GUI, the terminal is an application which can be opened by finding the terminal in the application viewer and clicking it. You will need to open a terminal to complete each of the challenges in this exercise.

In Linux, files have paths within the file system. The root location in the file system, the location from which all other file paths originate. In the Linux file system, the root location is indicated with the forward slash **(/)** character. The **/etc** directory is one of the directories one level deep from the root directory. One of the files in the **/etc** directory is the **shadow** file. This file contains encrypted formats of the passwords for users on the machine.

In order to navigate to directories in the Linux terminal, use the command **cd**. Follow the command with the file path of the target directory. For example, consider the file path: **/home/Desktop/Misc/random.txt**. In order to access the file, **random.txt**, one can first navigate to the directory containing the file, in this case, the **Misc** directory. To accomplish this, use the following command:

```
$ cd /home/Desktop/Misc
```

From there, one can view the contents of the file using the **cat** command. While in the **Misc** directory, one can view the contents of the **random.txt** file by running the command:

```
$ cat random.txt
```

Additionally, one can use the **cat** command with the file path rather than navigating to the directory first. For example, one could run the following command to display the contents of the **random.txt** file:

```
$ cat /home/Desktop/Misc/random.txt
```

This laboratory exercise has the flags hidden as contents of the files. In order to discover the flags, display the contents of the files.

Challenge 1

Use the command:

```
$ cat /etc/shadow
```

One of the lines in the file has the following contents:

"flag:$encryptedpasswordsarekeptin/etc/shadow".

Challenge 2

Use the command:

```
$ cat /bin/weirdfile
```

The contents of this file are: *"thingsarenotalwaysastheyappear"*.

Challenge 3

The flag for this challenge is spread across four files in the **/sbin** directory. These files are all fish themed, namely, *tilapia, salmon, bass,* and *hering*. By displaying the contents of these files, the flag is revealed. One can take advantage of the ability to chain commands to display the contents of all four files at one time. Use the command:

```
$ cat tilapia ; cat salmon ; cat bass ; cat hering
```

The flag revealed is: *"systembinariesaren'tfishy"*.

Challenge 4

The flag for this challenge is the name of the file where larger page sizes are handled, in the directory which contains information on device files. The directory in question is the **/dev** directory. The file is **hugepages**. The flag is the file name: *hugepages*

Challenge 5

The flag for this file is kept in the directory where process information is typically kept. In this case, the directory in question is the **/proc** directory. There is a specific file in this directory, the **partitions** file, which contains information on the partitions of the file system. The number of blocks in the **sda2** partition is the flag for this challenge.

Challenge 6

The flag for this challenge is contained in the directory where backups are kept, namely: **/var/backups**. The specific file in question is titled **apt.extended_states.0**. Display the contents of the file by using the command:

```
$ cat /var/backups/apt.extended_states.0
```

This will reveal the flag: *itsalwaysgoodtokeeparecordofwhatversionswereinstalled.*

Challenge 7

The flag for this challenge is actually missing. The directory which would contain the flag is where temporary files are kept, namely, the **/tmp** directory. The flag, which was stored there is removed because temporary files are deleted regularly. Rather, simply use the flag *tempfilesareremoved* for this challenge.

Challenge 8

The flag for this challenge is kept in a hidden file. Files can be listed in the Linux terminal using the command **ls**. However, files which are hidden (prefaced with the **.** symbol, ex: .hiddenfile.txt), will not be displayed using the basic **ls** command. In order to display hidden files, use the **-a** option with the **ls** command. Navigate to the directory where games are located, namely: **/usr/games**. This can be done using the command:

```
$ cd /usr/games
```

From there list all files using the command:

```
ls -a
```

This will display a file **.spaceinvaders**. Displaying its contents using the **cat** command will reveal the flag: *therearenogamesonthismachine!*

Challenge 9

The flag for this file can be found in the **/lib/ufw** directory. Navigate to the directory using the **cd** command:

```
$ cd /lib/ufw
```

The flag is kept in one of the files. The flag is *ufwissuitedforhostbasedfirewalls.*

Challenge 10

The flag for this challenge exists in the directory where removable devices are kept, namely **/media**. There is a file in this directory named **flag**. The contents are a ciphertext, meaning it is encrypted text. The type of cipher used is known as a vigenere cipher. The key for the encryption and decryption is the name of the file, **flag**. By decrypting the ciphertext with the key, the flag is revealed to be *thisiswhereusbdrivesappear.*

Chapter 6: Linux Hardening

In this laboratory exercise, the student will be introduced to the basics of Linux Hardening.

6.1 Laboratory Exercise

Linux Hardening

6.1.1 Specifications

The variant of Linux being used for this laboratory exercise is Ubuntu 16.04.6, an older version of the popular Ubuntu operating system. The machine been configured with numerous common vulnerabilities.

6.1.2 Learning Objectives

- Basics of Linux Hardening
- Basics of Linux Usage

6.1.3 Mapping to NIST Nice Framework

This laboratory exercise is intended to increase the student's familiarity with Linux. Additionally, the student will become familiar with common vulnerabilities and how to harden them. This laboratory exercise maps to the following KSAs from the NIST NICE Framework:

- Cybersecurity and Privacy Principles (K0004)

- Cyber Threats and Vulnerabilities (K0005)
- Basic System and OS Hardening Techniques (K0205)
- Recognizing Types of Vulnerabilities (S0078)
- System, Network, and OS Hardening Techniques (S0121)

6.1.4 Necessary Background and Expected Completion Time

This laboratory exercise can be completed by students with varying background and experience. The following categories should help identify approximately how much time (in minutes) will be necessary to complete the laboratory exercise, for a student meeting the criteria for the respective experience level.

- Beginner: A student in this category has little to no experience with Linux and the Linux terminal.
- Intermediate: A student in this category has experience with Linux and the Linux terminal but has little to no experience with Linux hardening.
- Advanced: A student in this category has experience with Linux and the Linux terminal as well as experience with Linux hardening.

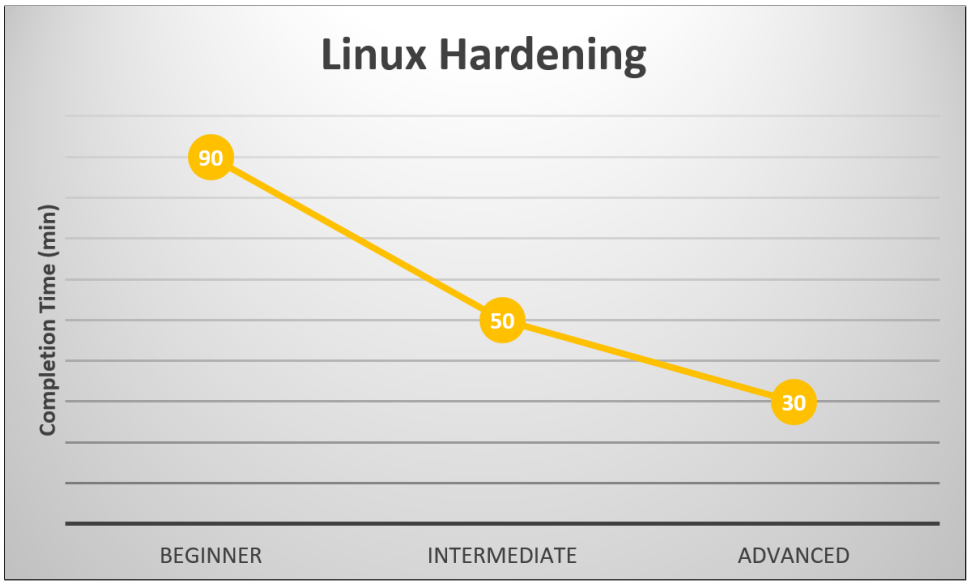

Figure 6.1: *Linux Hardening Laboratory Exercise Expected Completion Time (min)*

6.1.5 Configuration and Setup

The machine used in this laboratory is an installation of Ubuntu 16.04. It will be configured with the vulnerabilities listed in the vulnerability overview. This machine is configured with additional users, automatic updates disabled, weak passwords, erroneous cronjobs, and unprotected SSH enabled. The machine is configured using the initialization script, listing **6.1**

Listing 6.1: *lh-initializationscript.sh*

```
#!/bin/bash

#overhead
sudo dpkg --configure -a
sudo apt-get install git -y
sudo apt-get install openssh-server openssh-client -y
sudo service ssh start
sudo rm -r CYOTEE
sudo git clone https://github.com/CenterForSecureAndDependableSystems/CYOTEE.git

#create users
useradd redteam
useradd guest

#assign passwords to the users
sudo echo -e "redteam\nredteam" | passwd redteam
sudo echo -e "guest\nguest" | passwd guest

#disable auto-updates
sudo rm /etc/apt/apt.conf.d/20-auto-upgrades
sudo rm /etc/apt/apt.conf.d/20auto-upgrades
sudo cp CYOTEE/CYOTEE_Code_Linux/20-auto-upgrades /etc/apt/apt.conf.d/20-auto-
    upgrades

#add a couple of cron jobs

sudo crontab -u thesis -l | { cat; echo "* * * * * touch ~/Desktop/sensitivefile"
    ; } | crontab - -u thesis
sudo crontab -u thesis -l | { cat; echo "*/2 * * * * rm ~/Desktop/sensitivefile";
     } | crontab - -u thesis
```

6.1.6 Vulnerability List

1. Default, Weak, or Common Password
2. Additional Accounts
3. Disabled Automatic Updates
4. SSH
5. Cronjobs

6.1.7 Challenges

1. *Change Password*

   One of the most common vulnerabilities seen in machines and devices is password security. Many devices do not come preconfigured with a password, while others have a default

password. The first thing one should ensure is that their machine or device is password protected, otherwise it would be similar to not placing a lock on your front door, leaving yourself vulnerable to anyone. Once you have ensured you have a password on your machine, then consider: is it a secure password or not? Generally, a password should be changed from the default. Many individuals believe that because their device or machine has a default password, it is secure, however this is not correct. Default passwords are often applied to all devices from a similar batch and are frequently code-like PINs such as 0000 or 1234. For this reason, an individual should ensure that the default password is changed, otherwise it is as though you have bought a lock for your front door but everyone in the neighborhood has the same key which unlocks your door. Changing the default password does that mean your device is secure because your password may be on the list of common passwords. Each year a list of commonly used passwords is published by various organizations on the Internet. Common passwords include the word "password", reusing the username as the password, an empty password, and many others. An individual should consult these lists of common passwords to ensure that they have not accidentally and unknowingly used one of them. Once you have assigned a password, changed the default, and checked that your password is not on a list of commons ones, your password still may not be considered secure. Passwords also vary in what is known as their "strength", or how challenging it would be to crack. Common recommendations for ensuring high strength passwords include having a long length password, varying the characters in the password (uppercase, lowercase, special characters, numbers, etc), and avoiding a password containing personal information such as your pet's name. The default password on this machine is password.

For this task, create a high strength password and reset the current weak password to the new one.

2. *Remove/Disable Unnecessary Accounts*

When one uses a personal computers at home, they often only has one account on the machine: the one they set up themselves. Multi-user machines are a commonplace and often found in shared spaces such as schools, libraries, and cyber-cafés. For example, a

home computer may have two accounts, one for the parents and one for the children, with the childrens' account having less privilege and access than the adults' account. When acquiring a machine, or configuring a new machine, one should always check what user accounts are on that machine. While additional accounts are not inherently malicious, they may be providing an insecure backdoor into the machine. For this reason, one should ensure that unnecessary accounts are removed. If one is unsure whether the additional account is necessary or not, they can first disable or deescalate the privileges it has.

For this task, find the additional accounts that exist. First disable them, then delete/remove them.

3. *Enable Automatic Update Alerts*

   Updates are a very important element of computing. Computers and the software they run are often vulnerable, but updates provide patches for these vulnerabilities, so they can no longer be exploited. Unless update sites are checked daily to know when a new update is available, a critical update, providing a patch for a serious vulnerability, may accidentally be missed. New vulnerabilities are discovered frequently with exploits developed shortly thereafter. Manually checking for updates can result in missed updates which can leave the user susceptible to exploits. For this reason, it is important to have automatic updates enabled and to update regularly.

   For this task, enable the automatic updates on the machine.

4. *Disable or Harden SSH*

   Secure shell, better known as SSH, is a protocol which allows a user to log into a remote machine via a terminal. This can be useful for individuals who work remotely, or need to host a machine remotely. While this service can be useful, it also poses a security risk. Allowing remote access to your machine means that if a malicious user can bypass the password protection, that user now have full access to your machine. When it is not practical to disable SSH, one can improve the security of SSH by using SSH keys for authentication rather than a password.

   For this task, log into the vulnerable machine using SSH. Next, disable the

SSH service on the vulnerable Linux machine. Lastly, restart the SSH service and harden the service using SSH keys.

5. *Remove Unnecessary Cronjobs*

   Cron is a software utility which allows a user to schedule tasks to be performed at specified intervals. This is useful for tasks such as backing up a machine or checking for updates, among other tasks. Users can edit a Crontab file, which is a file containing the tasks which need to be run. These tasks are more commonly referred to as "Cronjobs". The Crontab file uses a specific format, allowing the user to specify the interval over which the Cronjob should be run using the metrics: minutes, hours, day of month, month of year, and day of week. The format for a Cronjob in the Crontab file is as follows:

   ```
   Min.  Hr.  DayOfMonth Month DayOfWeek Command
   ```

   Using an asterisk (*) in place of any of the fields in the format means that any value for that field will be accepted. For example, if you wanted to run the command `ls` 30 minutes after each hour, your formatted Cronjob would be:

   ```
   30 * * * * ls
   ```

   Other symbols, such as the step value symbol (/) can be used to provide further control. For example, to create a backup of your history every thirty minutes, your formatted Cronjob would be:

   ```
   */30 * * * * history > history.txt
   ```

   Cronjobs can be powerful but they can also be a covert way to perform some malicious action at fixed intervals, such as sending collected keystrokes to a remote machine. For this reason, one should regularly check their Crontab file to ensure that no unintended Cronjobs have been added.

   For this task, identify the unnecessary Cronjob running on this machine and remove the job from the Crontab file.

6.2 Solutions and Guided Walkthrough

6.2.1 Solutions

1. *Change Password*

   To change a user account password on a Linux machine, first open a terminal. After launching a terminal, type the command:

   `passwd username`

   This command will begin an interactive dialog within which you will enter the current password and then the new password.

2. *Remove/Disable Unnecessary Accounts*

   There are many ways to disable or lock accounts in Linux systems, some of which do not preserve the current password associated with the account to be locked. One of the methods for disabling a Linux user account, while preserving the current password, is to use the following command:

   `passwd username -l`

   Once ready to unlock the account, run the command:

   `passwd username -u`

   The **/etc/shadow** file contains information related to usernames and passwords. Locking and unlocking an account using the method above modifies the contents of the **/etc/shadow** file to indicate if the account is locked. To delete a user, use the command:

   `userdel username`

3. *Enable Automatic Update Alerts*

   To enable automatic update alerts on the Linux machine, click the "Settings" button at the top right corner of the screen. This will display a dropdown menu, select the "Systems Settings", which will display the systems settings GUI. Next, under the "System" tab, click on the "Software & Upgrades" icon. Next, select the "Updates" tab and set the options as desired with automatic update alerts enabled.

4. *Disable or Harden SSH*

To SSH onto the machine, run the command:

```
ssh <targetuser>@<target-ip>
```

To stop the SSH service from running, use the command:

```
sudo service ssh stop
```

You can try to SSH into the machine after performing this step to confirm that it is no longer possible. In addition to disabling SSH, one can remove the SSH from the machine all together by running the following two commands:

```
sudo apt-get purge openssh-server
```

```
sudo apt-get purge openssh-client
```

To harden SSH by using SSH keys, run the following commands:

```
ssh-keygen -t rsa
```

Enter the location to save the keys to, as well as a passphrase if one will be required. Next, copy the public key to the remote machine which should have SSH access. This is accomplished by running the command:

```
ssh-copy-id <remoteuser>@<remote-ip>
```

5. *Remove Unnecessary Cronjobs*

   To remove Cronjobs, edit the Crontab file by using the command:

   ```
   crontab -e
   ```

   Scroll through the file until you find the job you would like to remove. It can be removed by either commenting out the line by appending a pound symbol (#) or by deleting the line altogether. Save and exit the Crontab file and allow the new Crontab file to install.

6.2.2 Guided Walkthrough

In order to complete the challenges in this laboratory exercise, see the steps in this walkthrough.

A Linux workstation is a staple at cyber defense competitions and in the workplace. Ensuring that the workstation is secure is integral to ensuring that all of the data on the machine is secure. Without proper workstation security, an malicious user does not need to perform a complex attack to gain access to secure files, but rather can simply use regular methods of accessing the

machine to gain access.

Challenge 1

The first step that you should do upon being given a machine is to change the password. Passwords should not only be changed once, but relatively frequently. To change your password on a Linux machine, use the **passwd** command. In Linux, you can specify the user for whom you are trying to change the password. Run the following command to change the password for an example user named *sampleuser*:

```
$ passwd sampleuser
```

This will allow you to change the password.

Challenge 2

Occasionally, you may discover accounts on your machine which are either unnecessary (Guest) or malicious. It is important to assess the situation and determine whether it is appropriate to disable or remove the user. In the following cases, the target user will be named *sampleuser*. Run the following command to lock/disable a user account in Linux:

```
$ passwd sampleuser -l
```

If determined that the account does not need to be locked/disabled, unlock the account by running the following command:

```
$ passwd sampleuser -u
```

If determined that the account needs to be deleted all together, run the following command:

```
$ userdel sampleuser
```

Challenge 3

Updates a cornerstone in secure systems. Vulnerabilities are discovered every day and patches for these vulnerabilities frequently come soon after the vulnerability is discovered. Though updates are critical, it can be challenging to remember to check for updates. For this reason, having auto updates enabled can be helpful. In order to enable automatic updates on a Linux Desktop machine, navigate to the “Settings” button in the top right corner of the display. This will display a dropdown menu, select the “Systems Settings”, which will display the systems settings GUI. Next, under the “System” tab, click on the “Software & Upgrades” icon. Next, select the “Updates” tab and set the options as desired with automatic update alerts enabled.

Challenge 4

SSH is a service on Linux machines which allows remote access to a machine. This can be helpful at times, but also poses a significant security risk. You should always assess whether certain services are necessary on your machine or not. If you cannot easily determine whether SSH is necessary, you can disable the service by stopping it from running temporarily. If the service is necessary, steps can be taken to harden SSH.

To SSH onto the machine, run the command:

```
ssh <targetuser>@<target-ip>
```

To stop the SSH service from running, use the command:

```
sudo service ssh stop
```

You can try to SSH into the machine after performing this step to confirm that it is no longer possible. In addition to disabling SSH, one can remove the SSH from the machine all together by running the following two commands:

```
sudo apt-get purge openssh-server
```

```
sudo apt-get purge openssh-client
```

To harden SSH by using SSH keys, run the following commands:

```
ssh-keygen -t rsa
```

Enter the location to save the keys to, as well as a passphrase if one will be required. Next, copy the public key to the remote machine which should have SSH access. This is accomplished by running the command:

```
ssh-copy-id <remoteuser>@<remote-ip>
```

Challenge 5

Cronjobs are helpful in automating tasks. Although, they are also an attack vector which is rarely considered. Because cronjobs allow a job to be run at fairly infrequent intervals, you may not even notice that the job is running unless you check. You can check the list of cronjobs on your machine by using the following command:

```
$ crontab -e
```

This will open the crontab file. You can then scroll through all the jobs and look for erroneous cronjobs. If you are unsure whether a job is necessary or not, you can comment the job out

using the pound symbol (#). If a job is not needed, you can remove the line from the file all together.

Chapter 7: MySQL Usage & Hardening

In this laboratory exercise, the student will be introduced to MySQL hardening as well as familiarized with perform MySQL queries.

7.1 Laboratory Exercise

MySQL Hardening & Basics

7.1.1 Specifications

The variant of Linux being used for this laboratory is Ubuntu Server 16.04.6. For this lab, it is be used as a database server.

7.1.2 Learning Objectives

- Hardening a MySQL Database
- MySQL Basic Commands

7.1.3 Mapping to NIST Nice Framework

This laboratory exercise is intended to increase the student's familiarity with Linux and MySQL. The student should be familiar with accessing, managing, and querying a MySQL database. This laboratory exercise maps to the following KSAs from the NIST NICE Framework:

- Cybersecurity and Privacy Principles (K0004)

- Cyber Threats and Vulnerabilities (K0005)
- Data Administration (K0020)
- Database Management Systems (K0023)
- Query Languages (K0069)
- Basic System and OS Hardening Techniques (K0205)
- Database Theory (K0420)
- Generate Queries (S0037)
- Recognizing Types of Vulnerabilities (S0078)
- System, Network, and OS Hardening Techniques (S0121)
- Maintain Databases (A0176)

7.1.4 Necessary Background and Expected Completion Time

This laboratory exercise can be completed by students with varying background and experience. The following categories should help identify approximately how much time (in minutes) will be necessary to complete the laboratory exercise, for a student meeting the criteria for the respective experience level.

- Beginner: A student in this category has little to no experience using Linux and the Linux terminal. Additionally, this student has little to no experience using MySQL.
- Intermediate: A student in this category has experience with Linux and the Linux terminal but has little to no experience using MySQL.
- Advanced: A student in this category has experience with Linux, the Linux terminal and has experience using MySQL.

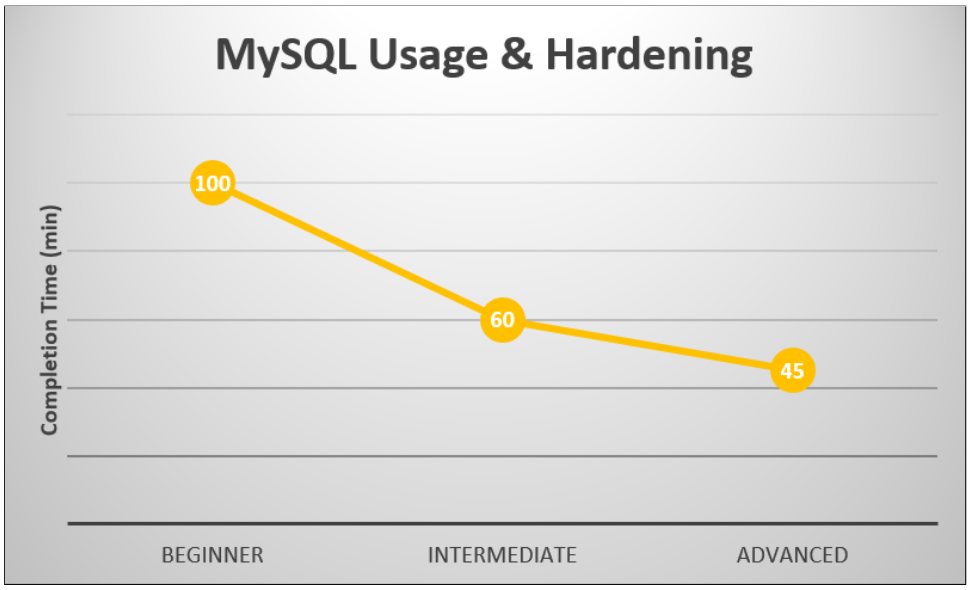

Figure 7.1: *MySQL Hardening & Basics Laboratory Exercise Expected Completion Time (min)*

7.1.5 Configuration and Setup

The machine in this laboratory exercise is an installation of CentOS 7. The machine has been configured with vulnerabilities listed in the vulnerability overview. The vulnerabilities include default or weak passwords, insecure MySQL root accounts, poorly configured MySQL permissions, default port usage, unnecessary databases and users, and automated startup tasks. MySQL is installed as a part of this laboratory exercise. The initialization script for this laboratory exercises reaches out to a GitHub repository to clone necessary files for the exercise. The initialization script also performs the creation and population of the MySQL databases and tables. The machine is configured using the initialization script, listing **7.1**.

Listing 7.1: *ms-initializationscript.sh*

```
#!/bin/bash

sudo dpkg --configure -a

#install git
sudo apt-get install git -y;

#remove existing repo and clone git repo
sudo rm -r CYOTEE
sudo git clone https://github.com/CenterForSecureAndDependableSystems/CYOTEE.git

#install MySQL
sudo apt-get install -y mysql-server;

#create the MySQL users
sudo mysql -u root -e "CREATE USER 'randomuser'@'localhost' IDENTIFIED BY '
    password'";
sudo mysql -u root -e "CREATE USER 'redteamer'@'localhost' IDENTIFIED BY 'redteam
    '";
sudo mysql -u root -e "CREATE USER 'haxxor'@'localhost' IDENTIFIED BY 'haxxor'";
sudo mysql -u root -e "CREATE USER 'testuser'@'localhost' IDENTIFIED BY 'test'";

#create the unnecessary database
sudo mysql -u root -e "CREATE DATABASE dontlook";

#remove the vulnerable database if one already exists
sudo mysql -u root -e "DROP DATABASE vulndb";

#create the vulnerable database
sudo mysql -u root -e "CREATE DATABASE vulndb";

#grant all privileges to users
sudo mysql -u root -e "GRANT ALL PRIVILEGES ON vulndb.* TO 'root'@'localhost'";
sudo mysql -u root -e "GRANT ALL PRIVILEGES ON vulndb.* TO 'testuser'@'localhost'
    ";

#import vulndb sql file
sudo mysql -u root vulndb < CYOTEE/CYOTEE_Code/SQL/vulndb.sql;
```

7.1.6 Vulnerability List

1. MySQL Root Account
2. MySQL Permissions
3. Default Ports
4. Unnecessary Databases and Users
5. Automated Startup Tasks

7.1.7 Challenges

1. *Secure MySQL Root Account*

   Root accounts on machines are typically the account which has the highest privileges. It is critical that the root account be properly secured as the root user can perform many super user tasks that a normal user may not have permission to perform. MySQL comes preconfigured with a root account which is not secured in the default installation of MySQL. Because of this, one is able to access the root account simply by typing the command:

   `$ mysql -u root`

   For this task, set the root account password to be something secure.

2. *Harden MySQL Permissions*

   MySQL allows permissions to be configured for tables in a database on a user-by-user basis. A few of the most commonly used MySQL commands are SELECT (equivalent to read permissions), INSERT, and DELETE. Databases being used by a web page to populate fields, for example, likely only need read access on the database and therefore should only have the SELECT privilege. The user **testuser** has all privileges on all tables in the database *vulndb*. This was accomplished by using the command:

   `GRANT ALL PRIVILEGES ON vulndb.* TO 'testuser'@'localhost';` In a later exercise, you will learn about tailoring permissions based on the concept of least privilege as needed in a web application.

   For this task, modify the privileges so that testuser has only read access on all the tables in the database *vulndb*.

3. *Change Default Ports*

   Network ports can be thought of as doors. Each application on a computer has its own port, or door, which is used for data to flow to and from that application. MySQL uses port 3306 by default, however, this can be changed. Although changing the port MySQL uses does not inherently improve security, it does defend against automated attacks which specifically target port 3306.

For this task, change the port that MySQL uses to a different port which is currently unused.

4. *Remove Unnecessary Databases or Users*

   Often, a base installation of MySQL will already include multiple MySQL users and a few example databases. These databases and users can be a vulnerability which attackers may exploit to gain access to your MySQL server because they are often not considered when securing the server and databases on it. It can be best to remove or disable these accounts and databases in order to avoid forgetting they exist, thereby forgetting to secure them.

   For this task, first find and remove the account which seems least likely to be a valid account on the machine and then find and remove the database which seems most likely to be associated with the unnecessary user.

5. *Using MySQL*

   MySQL commands are performed by using a specific syntax.

   For this task, perform the following steps:

   (a) Create a new database named `television`

   (b) Create a table in the newly created database named `shows` with the following fields (also called columns)

      i. `name`

      ii. `startyear`

   (c) Add another field named `endyear`

   (d) Create an entry for the following television shows in the table

      i. Boy Meets World

      ii. That 70's Show

      iii. Saved by the Bell

   (e) Remove the entry for Boy Meets World

   (f) Add an entry for the show Girl Meets World

Performing these tasks should provide you with a basic understanding of using MySQL and enable you to create and modify a database.

6. *Querying MySQL Databases*

   Often, rather than being tasked with creating and managing a database, one will be asked to find data in a table. Various actions such as reading, inserting, and deleting data to and from a database are known as queries. Reading information from a table is done in MySQL by using the SELECT command along with specific parameters if the target data is known.

   For part A of this task, you will be querying the table *useraccounts* in the database *vulndb* to read the data.

   Part B of this task requires that you query the table *employees* in the database *vulndb* to read the data. Keep in mind that queries can be made where field values are specified, and multiple field-value pairs may be chained together for a more specific query.

   (a) Query to find the solutions to the following questions.

      i. What is the name of the individual whose locations is New York?

      ii. What is the salary of Jeremiah Houston?

      iii. What is the name of the individual whose location is Los Angeles and has a salary of $500,000?

   (b) Query to find the solutions to the following questions.

      i. How old is Tami Vasquez?

      ii. What does Wanda Lloyd do for work?

      iii. What are the occupations of the 40 year-olds?

7.2 Solutions and Guided Walkthrough

7.2.1 Solutions

1. *Secure MySQL Root Account*

   MySQL is an application which has the capability for a user to harden the root account

and set a strong password by running the command:

`sudo mysql_secure_installation`

This will begin an interactive dialog in which you will set the MySQL root account password.

2. *Harden MySQL Permissions*

   To ensure **testuser** has read-only access on all tables in the *vulndb* database, run the following MySQL command:

   `GRANT SELECT ON vulndb.* TO 'testuser'@'localhost';`

   To ensure that this permission change takes place, run the following command:

   `FLUSH PRIVILEGES`

   This will reload the privileges table.

3. *Change Default Ports*

   To change the default port which MySQL uses, open the file located at `/etc/my.cnf` and change the line which indicates which port is being used or add a line `port = <xxxx>` if one does not already exist.

4. *Remove Unnecessary Databases or Users*

   You can check the list of users on a MySQL server by running the MySQL command:

   `SELECT user, host FROM mysql.user;`

   Here you will find a user named `haxxor`, that is the malicious user. To remove a user, run the MySQL command:

   `DROP USER '<username>'@'<host>'`

   By navigating through the databases, you will find one titled *dontlook*. This database is empty and is the unnecessary table associated with this task. To remove a table, run the MySQL command:

   `DROP TABLE <tablename>.`

5. *Using MySQL*

   For this task, reference the list of commands below, they should be sufficient to help you perform the tasks.

- Create a Database

  ```
  CREATE DATABASE <dbname>;
  ```

- List all Databases

  ```
  SHOW DATABASES;
  ```

- Enter a Database

  ```
  USE <dbname>;
  ```

- List all Tables in a Database

  ```
  SHOW TABLES;
  ```

- Create a Table

  ```
  CREATE TABLE <name> (<field_1 type_1,...,field_n type_n>);
  ```

- Create a Table Entry

  ```
  INSERT INTO <name> (<field_1,...,field_n>) VALUES (<value_1,...,value_n>);
  ```

- Delete an Entry

  ```
  DELETE FROM <name> WHERE <field> = <value>;
  ```

- Add a Field to a Table

  ```
  ALTER TABLE <name> ADD COLUMN <col_name col_type>;
  ```

6. *Querying MySQL Databases*

   Queries in MySQL can be formatted in many ways. A common way to get results matching a query is to the following syntax:

   ```
   SELECT <fields> FROM <name> WHERE <field> = <value>;
   ```

   Answers:

   - Jeanette Wise lives in New York.
   - Jeremiah Houston's salary is $50,000.
   - Emma Castillo makes $500,000 in Los Angeles.
   - Tami Vasquez is 12 years old.
   - Wanda Lloyd works as a Barista.
   - The 40 year-olds are a teacher and a laywer.

7.2.2 Guided Walkthrough

In order to complete the challenges in this laboratory exercise, see the steps in this walkthrough.

Databases are at the heart of many organizations. They allow for organized, electronic record storage. Additionally, using a database management system such as MySQL, you can perform powerful queries to manipulate specific data. Understanding how to manipulate the data in a database using MySQL and ensuring that your MySQL server is secure are critical.
In MySQL, databases are comprised of tables which hold records (data entries). A table is defined with specific fields (columns) of data which the individual records, or entries, will have. You can access the MySQL prompt by entering the following command in a terminal for a user named *sampleuser*:

```
$ mysql -u sampleuser
```

Challenge 1

A default installation of MySQL leaves the root user with no password, allowing anyone to access the databases with root privileges. You can set a password for the root user by running the following command:

```
$ sudo mysql_secure_installation
```

This will begin an interactive dialog in which you will set the MySQL root account password.

Challenge 2

MySQL users can have many different privileges on various databases in the system. MySQL allows you to configure what databases and tables specific users have certain privileges on. For this challenge, you are tasked with ensuring that **testuser** has read-only access on all tables in the *vulndb* database. To do this, enter the following command in the MySQL prompt:

```
GRANT SELECT ON vulndb.* TO 'testuser'@'localhost';
```

In this command, SELECT is the name of the privilege which will be granted, vulndb is the database which the privilege will be granted on, the asterisk (*) specifies that all tables in the preceeding database should be affected, and 'testuser'@'localhost' is the name and host of the user who will be affected by this granting of privileges. To ensure that the permission change

takes place, run the following command in MySQL:

```
FLUSH PRIVILEGES;
```

This will reload the privileges table.

Challenge 3

The various services on a machine run on different ports. By default, MySQL runs on port 3306 (MySQL traffic goes in and out of port 3306). It can be helpful to change the default port to something else. To change the port, open the file located at **/etc/my.cnf**. In this file, find the line which specifies the port being used. The line should read: **port 3306**. If the line does not already exist, add the line, changing 3306 to something else. Be sure that the new port is not already in use.

Challenge 4

MySQL has various users who have different accesses to the data in a MySQL server. Ensuring that only necessary users have access to the data, follow the instructions below.

You can check the list of users on a MySQL server by running the MySQL command:

```
SELECT user, host FROM mysql.user;
```

Here you will find a user named `haxxor`, that is the malicious user. To remove a user, run the MySQL command:

```
DROP USER '<username>'@'<host>';
```

By navigating through the databases, you will find a table titled *dontlook*. This table contains just ID numbers and is the unnecessary table associated with this task. To remove a table, run the MySQL command:

```
DROP TABLE <tablename>;.
```

Challenge 5

For this task, reference the list of commands below, they should be sufficient to help you perform the tasks.

- Create a Database

  ```
  CREATE DATABASE <dbname>;
  ```

- List all Databases

```
SHOW DATABASES;
```

- Enter a Database

  ```
  USE <dbname>;
  ```

- List all Tables in a Database

  ```
  SHOW TABLES;
  ```

- Create a Table

  ```
  CREATE TABLE <name> (<field_1 type_1,...,field_n type_n>);
  ```

- Create a Table Entry

  ```
  INSERT INTO <name> (<field_1,...,field_n>) VALUES (<value_1,...,value_n>);
  ```

- Delete an Entry

  ```
  DELETE FROM <name> WHERE <field> = <value>;
  ```

- Add a Field to a Table

  ```
  ALTER TABLE <name> ADD COLUMN <col_name col_type>;
  ```

To create a new database named **television**, use the following command:

```
CREATE DATABASE television;
```

To begin running commands within that database, use the following command:

```
USE television;
```

To create a table named **shows** with fields **name** and **startyear**, use the following command:

```
CREATE TABLE shows (name VARCHAR(255), startyear INT);
```

To add another field named **endyear** to the **shows** table, use the following command:

```
ALTER TABLE shows ADD COLUMN endyear INT;
```

To create the entries for the designated television shows, use the INSERT INTO command.

To remove the entry for Boy Meets World, use the DELETE FROM command.

Challenge 6

Queries in MySQL can be formatted in many ways. A common way to get results matching a query is to the following syntax:

```
SELECT <fields> FROM <name> WHERE <field> = <value>;
```

Answers:

- Jeanette Wise lives in New York. Query for location = “New York”.
- Jeremiah Houston’s salary is $50,000. Query for name = “Jeremiah Houston”.
- Emma Castillo makes $500,000 in Los Angeles. Query for location = “Los Angeles” and salary = “500000”
- Tami Vasquez is 12 years old. Query for name = “Tami Vasquez”.
- Wanda Lloyd works as a Barista. Query for name = “Wanda Lloyd”.
- The 40 year-olds are a teacher and a lawyer. Query for age = “40”.

Chapter 8: Creating a Vulnerable Web Application

In this laboratory exercise, the student will be introduced to the concepts surrounding web applications and will create a vulnerable web application.

8.1 Laboratory Exercise

Web Application Creation

8.1.1 Specifications

This lab will be completed on an Ubuntu Server LTS 16.04.6 machine running Nginx version 1.14.0.

8.1.2 Learning Objectives

- Understand basics of web servers
- Configure a secure web server
- Create a web application

8.1.3 Mapping to NIST Nice Framework

This laboratory exercise is intended to increase the student's familiarity with webpages, web serving, and the interaction between PHP, MySQL, and Nginx. The student will need this information to successfully complete the web application hardening exercise. This laboratory exercise maps to the following KSAs from the NIST NICE Framework:

- Cybersecurity and Privacy Principles (K0004)
- Cyber Threats and Vulnerabilities (K0005)
- Application Vulnerabilities (K0006)
- Data Administration (K0020)
- Database Management Systems (K0023)
- Programming Language Structures and Logic (K0068)
- Query Languages (K0069)
- Application Security Threats and Vulnerabilities (K0070)
- Database Access Applicaiton Programming (K0197)
- Database Theory (K0420)
- Conducting Queries (S0013)
- Generate Queries (S0037)
- Writing Code (S0060)
- Develop Secure Software (A0047)
- Maintain Databases (A0176)

8.1.4 Necessary Background and Expected Completion Time

This laboratory exercise can be completed by students with varying background and experience. The following categories should help identify approximately how much time (in minutes) will be necessary to complete the laboratory exercise, for a student meeting the criteria for the respective experience level.

- Beginner: A student in this category has little to no experience with Linux, the Linux terminal, or Nginx.

- Intermediate: A student in this category has experience with Linux and the Linux terminal but has little to no experience with Nginx or web serving concepts.
- Advanced: A student in this category has experience with Linux and the Linux terminal. Additionally, this student has experience with web service concepts.

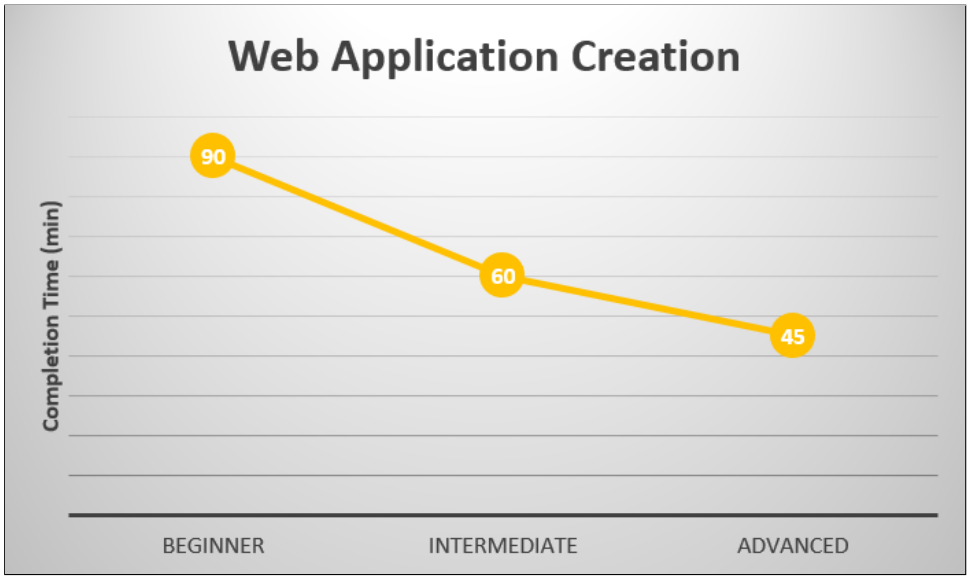

Figure 8.1: *Web Application Creation Laboratory Exercise Expected Completion Time (min)*

8.1.5 Configuration and Setup

The machine used in this laboratory exercise is an installation of Ubuntu Server LTS 18.04.1. Additionally, Nginx version 1.14.0 is installed on the machine to serve the webpage. The initialization script also installs Nginx, PHP, and MySQL. The machine is configured using the initialization script, listing **8.1**.

Listing 8.1: *vw-initializationscript.sh*

```
#!/bin/bash

sudo dpkg --configure -a


#install nginx
sudo apt-get install nginx -y

#install php
sudo apt-get install php-fpm php-mysql -y

#install MySQL
sudo apt-get install mysql-server -y
```

8.1.6 Challenges

1. *Stop Serving Page*

   Often times you will need to perform some sort of maintenance on your web server. Typically, this means you will need to stop serving your webpage for a brief moment.

   For this task, stop and then restart the web server. Verify that stopping the web server by trying to visit the site while the server is stopped.

2. *Identify Key File Locations*

   Websites are made up of files on a web server. The most common file is the `index.html` file which is typically the landing page when a user visits your website. There are a few other locations where critical files are kept. Identifying these locations will allow you to successfully configure your web server.

   For this task, identify the directory location for the following elements of your web server:

   - Web Page Content - that is the actual web pages which are being served and all the files associated with them.
   - Web Server Configuration - that is where all the configuration files for your web server are located.

3. *Enable HTTPS*

   Web servers often have users input usernames and passwords and other sensitive data. For

this reason, it is very important to ensure that the data being sent to and from a web server are secure. In order to secure this data in transit, one can use HTTPS. In order to properly configure HTTPS, there are a few different things which must be accomplished. First, the user must generate SSL keys and a certificate. Once they have done this, the web server must be configured to use those keys and certificate. In a real enterprise environment, the certificate would be validated by what is known as a Certificate Authority (CA) but for now, it can be left as a self-signed certificate.

For this task perform the following:

- Use OpenSSL to generate a certificate and key.
- Configure Nginx to listen on port 443.
- Configure Nginx to use the certificate and key.

This can be a challenging task to accomplish as the necessary syntax is very specific. You should use the internet to complete this task and see the solutions and/or the guided walkthrough if needed.

4. *Disable Unnecessary HTTP Methods*

   HTTP is a protocol which defines how to format messages to navigate the internet. There are many HTTP methods, but only a few are commonly used. The three most commonly used methods are GET, POST, and HEAD. Other than these methods, there are multiple others; specifically, two methods DELETE and TRACE leave web servers vulnerable. The DELETE method can be used to delete a specified resource from a web server similar to the way that the GET method retrieves a resource. The TRACE method can be used to identify what resources are being transmitted to the other end of a request chain. Typically, this information would be used for debugging purposes but provides an attacker the opportunity to trace where traffic is flowing.

   For this task, disable all HTTP methods other than the GET, HEAD, or POST methods. Once again, for this task, the internet is a good resource. Refer to the solution manual if needed.

5. *Prevent Giving Away Server Content Details*

   HTTP error codes are a common way of conveying what went wrong when an error occurs. Common error codes that most users may recognize are:

   - 401, unauthorized access
   - 404, resource not found
   - 403, permission denied
   - 405, method not allowed

   These error codes, while convenient, can convey a lot of detail about your directory structure and your data to an attacker. Unless it is absolutely critical that you let the user know what type of error occurred, it likely suffices to just let them know that an error occurred. For example, if the attacker receives a 403 error code, they will know that content exists in that location and that the content is likely of some value since it is being protected.

   For this task, route error 401, 403, 404, and 405 error codes to the same page that a 404 error routes to by default. Once again, make use of the internet and reference the solution manual as necessary.

6. *Creating a Web Application*

   For this challenge, follow along with the code provided from the GitHub repository to create the files for the vulnerable web application. Try to identify which lines in the code correspond to the tasks in this challenge. Running the initialization script for the web application hardening laboratory exercise, listing 9.1, will perform the tasks in this challenge. If you use the initialization script, listing 9.1 to configure the vulnerable web application, discuss with peers or colleagues how you would have approached each task if starting from scratch.

   - Ensure that the machine has Nginx, MySQL, and PHP installed.
   - Edit configuration files so the landing page is index.php.
   - Edit configuration files so that Nginx interfaces with PHP, allowing the server to serve PHP pages.

- Create a MySQL database with the following data:
  - Database named **test**.
  - Table named **employees** with fields **name** and **password**, both of type VARCHAR.
  - Table named **startdates** with fields **name** and **date**, both of type VARCHAR.
  - Input the following (name,password) pairs into the employees table:
    (liam,password)
    (emma,superman)
    (william,mustang)
    (sophia,trustno1)
    (mason,hockeymason)
    (mia,curiousgeorge).
- Create an **index.html** with the following specifications:
  - Accept a username input
  - Accept a password input
  - Include a submit button
  - Send the inputs to **auth.php**
- Create an **auth.php** with the following specifications:
  - Interact with MySQL database using *root* user.
  - Check if user credentials were correct.
  - Display a link to **insertdate.php?name=$n**, where $n is the name of all users matching the query.
  - If no matches, send to **invalid.html**.
- Create an **insertdate.php** with the following specifications:
  - Accept a username input
  - Accept a startdate input
  - Include a submit button

  - Send the inputs to **insert.php**

- Create an **insert.php** with the following specifications:
  - Interact with MySQL database using *root* user.
  - Insert an entry into the MySQL **startdates** table with the **name** and **startdate** fields set to the values received from the **insertdate.php** page.
  - Link back to **index.html**.
- Create an **invalid.html** which states invalid input and links back to the **index.html** page.

8.2 Solutions and Guided Walkthrough

8.2.1 Solutions

1. *Stop Serving Page*

   In order to stop serving webpages on a web server, run the following command from a terminal:

   ```
   sudo systemctl stop nginx
   ```

   To start serving the page again, run the following command from a terminal:

   ```
   sudo systemctl start nginx
   ```

2. *Identify Key File Locations*

   For this task, the two key file locations are:

   - Web Page Content - `/var/www/html`
   - Web Server Configuration - `/etc/nginx`

3. *Enable HTTPS*

   In order to enable HTTPS on your web server, there are a few steps which need to be followed:

   - Use OpenSSL to generate a certificate and key

     To perform this task, run the following command:

Listing 8.3: *Enabling SSL*

```
server{
    listen 443 ssl;
    server_name <name>;
    ssl on;
    ssl_certificate <path to certificate>;
    ssl_certificate_key <path to key>;
}
```

`openssl req -x509 -newkey rsa:4096 -keyout public.key -out certificate.cert -days 365`

You will be prompted for a passphrase, chose a passphrase which is easy to remember and store it securely.

- Configure Nginx to listen on port 443

  To complete this task, navigate to the directory `/etc/nginx/conf.d` and edit the file **ssl.conf**. Find the `server` block in the configuration file (**/etc/nginx/conf.d**) and add content to match Listing **8.2**:

Listing 8.2: *Enabling HTTPS*

```
server{
    listen 443 ssl;
    ...
}
```

- Configure Nginx to use the certificate and public key generated

  To accomplish this task, edit the file at `/etc/nginx/conf.d/ssl.conf` again and ensure that it matches the content in listing **8.3**:

4. *Disable Unnecessary HTTP Methods*

   In order to disable the HTTP methods excluding GET, HEAD, or POST, insert the segment seen in listing **8.4** into the file located at `/etc/nginx/conf.d/default.conf`:

5. *Prevent Giving Away Server Content Details*

   In order to reroute the 401, 403, 404, and 405 error codes to the default page for 404, insert the following line into the file located at `/etc/nginx/sites-enabled/default`:

   `error_page 401 403 404 405 /404.html`

Listing 8.4: *Disabling Unnecessary HTTP Methods*

```
if($request_method !~ ^(GET|HEAD|POST)$$)
{
    add_header Allow "GET, HEAD, POST" always;
        return 405;
}
```

6. *Creating a Web Application*

   See the walkthrough for detailed steps on this task.

8.2.2 Guided Walkthrough

In order to complete the challenges in this laboratory exercise, see the steps in this walkthrough. This guided walkthrough will outlines the steps when using VMware as the hypervisor, though VirtualBox can also be used and has similar functionality.

Creating the Virtual Machine

In order to begin this exercise, you will need to have access to a virtual machine running Ubuntu 16.04 Desktop as the operating system. If you do not already have access to a virtual machine running Ubuntu 16.04 Desktop, creating your own is a relatively simple. The directions from here forward are for a machine running Windows 10, though similar steps can be followed for other OSs.

Installing VMware

If an organization has the resources to purchase a license for VMware [41] Workstation Pro, they should do so. VMware workstation has extended capabilities including taking snapshots, creating and managing encrypted VMs, customizing virtual networks, and virtual machine cloning [24]. Although these extended capabilities can be convenient, everything necessary for this exercise can be performed with the free VMware Player. In order to download VMware Player, visit the VMware website and navigate to the Downloads tab. From the Downloads tab, navigate to VMware Workstation Player. From here, select the download button for Windows. The following link is valid at the time of creation (Setptember 2019): `https://www.vmware.com/products/workstation-player/workstation-player-evaluation.html` [32].

After downloading the installer, follow the steps that the installation platform provides. A webpage created by Shailesh Jha [33] contains a walkthrough of the installation process: **https://www.shaileshjha.com/step-by-step-install-vmware-workstation-player-12-in-windows-10/**.

Downloading Ubuntu 16.04 ISO

Although this exercise could be performed on many different OSs, the one chosen is Ubuntu in the interest of the development teams familiarity with Ubuntu for web development. In order to create a virtual machine running Ubuntu, the ISO file will need to be downloaded. In order to download the ISO, visit the Ubuntu website. On the website, navigate to the Download tab. In the Download tab, click on the Older Releases under the *Other Ways to Download* section. Next, click the link with Name: 16.04. Depending on whether your host machine is 64-bit or 32-bit, select the appropriate download link for the Desktop image. The download should begin automatically. At the time that this document is being created, the following link will take you to the download page: `http://releases.ubuntu.com/16.04/` [34].

Creating a Virtual Machine

In order to create a new virtual machine, run the VMware Workstation Player application you installed earlier. One of the options should be to *Create a New Virtual Machine*. Select that option and then the *New Virtual Machine Wizard* will appear. Radio boxes will appear asking where to install the operating system from, select the option which says *Installer disc image file (iso)* and then browse to wherever you downloaded the Ubuntu 16.04 ISO file. Select the next button.

Create a username and password for the machine. It does not matter what you choose as long they are credentials which you will have access to later on to log in. Select the next button.

Give the virtual machine a name. This is the name that will appear in your VMware Workstation Player GUI. Also specify where you would like the virtual machine stored. Select the next button.

Specify the Disk Capacity. This section can be left with the default values. Select the next

button.

If you are satisfied with everything up to this point, select the finish button. You can otherwise customize the hardware before finishing, although it is not necessary. You should now have a new virtual machine.

Installing Ubuntu on the Virtual Machine

At this point, either the virtual machine powered on automatically and you are viewing the console to it or you will have to power the machine on in the VMware Workstation Player GUI. Once the virtual machine is running, Ubuntu should begin installing on its own using the EasyInstall. If not, follow the steps in the installation process. A webapge created by Krishna [35] provides an in-depth guide on installing Ubuntu in VMware with images: `https://www.maketecheasier.com/install-ubuntu-in-vmware-player-windows/`.

Setting Up the Web Server

Setting up the Ubuntu virtual machine to be a web server requires a few steps. Follow the instructions to have a web page capable of serving PHP content and interacting with a MySQL database.

Installing Nginx

In order to install Nginx, open a terminal. This can be accomplished by selecting the Search Computer for Applications icon on the task bar. Search for **Terminal** and then select the Terminal icon to fire it up. Alternatively, pressing Ctrl + Alt + T will open a terminal. Type the following command to install Nginx:

```
$ sudo apt-get install nginx
```

To verify that Nginx was properly installed, open a browser (Firefox comes preinstalled on Ubuntu 16.04). In the URL bar, type **localhost**. If the installation was performed properly, you should be greeted by a *Welcome to nginx* page.

Installing PHP

At this point, your web server can serve HTML pages, but is not capable of using PHP which is integral to allowing server side capabilities to a web server. Open a terminal (see above for details on how to do this). At the terminal, type the following command:

```
$ sudo apt-get install php-fpm php-mysql
```

This will install PHP on your machine. Although PHP is installed, the step is not complete; you must now tell Nginx how to use PHP. This is done by editing the file located at /texttt/etc/nginx/sites-available/default. Type the following command in the terminal in order to edit the file:

```
$ sudo nano /etc/nginx/sites-available/default
```

There are a few locations in the file which must be changed. In the server block, there is a line specifying file names for the index file (the landing page). You should see the following line:

```
index index.html index.htm index.nginx-debian.html;
```

Add the text index.php to the beginning of the list of index file names so that the line now appears:

```
index index.php index.html index.htm index.nginx-debian.html;
```

The next location which must be edited is in the sub block which is labeled `location` $\sim$ `\.php$`. By default, this sub block is commented out. Remove the comments necessary so that the following lines are no longer commented.

```
location ~ \.php$

include snippets/fastcgi-php.conf;

fastcgi-pass unix:/var/run/php/php7.0-fpm.sock;
```

At this point, Nginx should be configured properly to serve PHP content. In order to test this, navigate to the `/var/www/html` directory. Here, create a new file called **index.php**. Do this by running the following command in the terminal:

```
$ sudo nano /var/www/html/index.php
```

Insert the following text into the file:

```
<?php
```

```
    phpinfo();
?>
```

Save the file and exit. Visit the web browser again. Once again, type **localhost** into the URL bar. If everything was done correctly, you should be served a page with all the information about your PHP installation such as the version. If this is not the case, please retry the steps ensuring that you have correctly performed the installation and file configurations [31].

Installing MySQL

Your web server should now be able to serve PHP content. The next step is to install MySQL so that the PHP pages can interact with MySQL databases. In order to install MySQL, run the following command in the terminal:

```
$ sudo apt-get install mysql-server
```

Done properly, the installation should begin and a prompt will appear asking you to set a password for the MySQL root user. As described in the prompt, this step is not mandatory, but strongly recommended. Set a strong password because the root user has full privileges to all tables in all databases. After entering the password, the installation process will continue. To test that MySQL was installed properly, enter the following command in the terminal:

```
$ mysql -u root -p
```

You will be prompted for the MySQL root user password which was set previously. After entering the password, you will be presented with a MySQL console. From here you can perform any MySQL commands including but not limited to [31]:

- Creating/Deleting users
- Modifying user privileges
- Creating/Deleting databases
- Creating/Deleting tables
- Entering data into tables

Creating the Vulnerable Application

Provided that you have not run into any errors up to this point, you are ready to create the vulnerable application. Prior to creating the files for the application, you will need to make some modifications in MySQL. Enter into the MySQL console by entering the following command in the terminal:

```
$ sudo mysql -u root -p
```

For the purposes of this tutorial, a weak password, namely: **sql**, has been selected for readability purposes.

After entering the password, you will be presented with a MySQL console. Once here, create a new database. You can name this database whatever you would like, but to follow along with the code presented later, name it **test**. You can do this by entering the following command in the MySQL console:

```
mysql> CREATE DATABASE test;
```

Next you will need to create a couple of tables. The table names do not have to be specific, but following the commands as written is most compatible with the code presented later. If you choose to use your own names, ensure that they are changed in the code later accordingly. To create the tables enter the following commands in the MySQL console:

```
mysql> USE test;
```

This command will change the active database to your newly created database.

```
mysql> CREATE TABLE employees (name VARCHAR(255), password VARCHAR(255));
```

This command creates a new table in the **test** database called **employees**. The fields, or columns, in this table are **name** and **password**, each of type VARCHAR with up to 255 characters (effectively a 255 character string).

```
mysql> CREATE TABLE startdates (name VARCHAR(255), date VARCHAR(255));
```

This command creates a new table in the **test** database called **startdates**. The fields, or columns, in this table are **name** and **date**, each of type VARCHAR with up to 255 characters (effectively a 255 character string).

Next, you will need to enter data into the **employees** database. Follow the command below

Listing 8.5: *vw-index.html*

```
<html>
	<body style="background-color: black; color: white">
		<div style="text-align: center">
			<form action="vw-auth.php" method="get">
				Username: <input type="text" name="user"><br>
				Password: <input type="text" name="pass"><br>
				<input type="submit">
			</form>
		</div>
	</body>
</html>
```

to enter the data:

```
mysql> INSERT INTO employees (name,password) VALUES ("<username>","<password>");
```

Replace the username and password values with usernames and passwords you would like. Enter the data for six users in this manner. You may use whatever names and passwords you like, but to follow along with what was entered during development, use the following commands:

```
mysql> INSERT INTO employees (name,password) VALUES ("liam","password");
mysql> INSERT INTO employees (name,password) VALUES ("emma","superman");
mysql> INSERT INTO employees (name,password) VALUES ("william","mustang");
mysql> INSERT INTO employees (name,password) VALUES ("sophia","trustno1");
mysql> INSERT INTO employees (name,password) VALUES ("mason","hockeymason");
mysql> INSERT INTO employees (name,password) VALUES ("mia","curiousgeorge");
```

Now that MySQL is configured as needed, you are ready to move onto writing the code necessary to create the vulnerable web application. For this section, you can either follow along with the explanation of the code. Create the following files in the `/var/www/html` directory.

index.html

This file, vw-index.html, listing **8.5**, contains a simple form into which a user will enter a username and password. Upon submitting their input, the contents of the text boxes will be passed to a page named **vw-auth.php** using the HTTP GET method.

Listing 8.6: *vw-auth.php*

```php
<?php

$servername = "localhost";
$username = "root";
$password = "sql";
$dbname = "test";

$conn = new mysqli($servername,$username,$password,$dbname);

$sql = "SELECT * FROM employees WHERE name='" . $_GET["user"] . "' AND password='
    " . $_GET["pass"] . "';";

$result = $conn->query($sql);

$rows = array();

if($result->num_rows > 0)
{
        while($row = $result->fetch_assoc())
        {
                array_push($rows,$row);
        }
        print "Click your name to access your date entry form!<br>";
        foreach($rows as $r)
        {
                $n = $r['name'];
                print "<a href='vw-insertdate.php?name=$n'>" . $n . "</a><br><br>";
        }
}
else
{
        header("Location: vw-invalid.html");
}

?>
```

vw-auth.php

In this file, vw-auth.php, listing **8.6**, variables with the servername, username, password, and database are used to establish a connection with the MySQL database. Next the values entered into the username and password fields on the **vw-index.html** page are used in a SQL query, the variable for this query is named **$sql**. The SQL query is then performed and the result is stored in the **$result** variable. Next, a check is performed to see if any matches were return (in theory a match is only returned if the correct username and password are entered). If matches were not returned, the user is served the **vw-invalid.html** page. If matches were returned, then the code loops over all the matches and creates a link to the **vw-insertdate.php** page for the associated user.

Listing 8.7: *vw-insertdate.php*

```
<?php
        $n = $_GET['name'];
?>

<html>
        <body style="background-color: black; color: white">
                <div style="text-align: center">
                        <form action="vw-insert.php" method="get">
                                Enter a date in MM/DD/YYYY format<br>
                                <input type="text" name="date"><br>
                                <input type="text" name="user" value="<?php echo $n;
                                    ?>" readonly><br>
                                <input type="submit">
                        </form>
                </div>
        </body>
</html>
```

vw-insertdate.php

In this file, vw-insertdate.php, listing **8.7**, the name of the user clicked on the **vw-auth.php** page is passed via the HTTP GET method. Then an HTML page is created which allows the user to enter a date into a text field. The requested format is MM/DD/YYYY. Another text field is auto populated with the username of the associated user; this field is read-only. Upon submission, the values in both text fields are passed to the **vw-insert.php** page via the HTTP GET method.

vw-insert.php

In this file, vw-insert.php, listing **8.8**, variables with the servername, username, password, and database are used to establish a connection with the MySQL database. Next the values of the username and date entered into the text fields on the **vw-insertdate.php** are stored in variables. There is a check to ensure that the connection to the database was successful. Next, a SQL query is constructed to insert into the **startdates** table. The query is constructed so the username and date passed to the page via the HTTP GET method will be inserted into a row of the table. The query is then performed. Finally, the user is presented with an HTTP page which states that the insertion has been completed and provides a link back to the logon page.

Listing 8.8: *vw-insert.php*

```
<?php

$servername = "localhost";
$username = "root";
$password = "sql";
$dbname = "test";

$user = $_GET['user'];
$date = $_GET['date'];

$conn = new mysqli($servername,$username,$password,$dbname);

if($conn->connect_error)
{
	die("Connection Failed: " . $conn->connect_error);
}

$sql = "INSERT INTO startdates (name, date) VALUES (\"$user\",\"$date\");";

$conn->query($sql);

?>

<html>
	<body style="background-color: black; color: white">
		<div style="text-align: center">
			<h1>Insert Complete!</h1>
			<button onclick="location.href='vw-index.html'">Return to
			    Login Page</button>
		</div>
	</body>
</html>
```

Listing 8.9: *vw-invalid.html*

```
<html>
	<body style="background-color: black; color: white">
		<div style="text-align: center">
			<h1>Invalid Logon Attempt!</h1>
			<button onclick="location.href='vw-index.html'">Return to
			    Login Page</button>
		</div>
	</body>
</html>
```

vw-invalid.html

In this file, vw-invalid.html, listing **8.9**, a simple HTML page is presented to the user. The page states that the logon attempt was invalid and provides a link back to the logon page.

Chapter 9: Web Application Hardening

In this laboratory exercise, the student will be introduced to common web application vulnerabilities and will mitigate against the vulnerabilities.

9.1 Laboratory Exercise

Web Application Hardening

9.1.1 Specifications

The variant of Linux being used for this laboratory is Ubuntu 16.04.6, an older version of the Ubuntu operating system. This machine has been configured with a vulnerable web application.

9.1.2 Learning Objectives

- Understand web applications
- Familiarize with web server applications such as Nginx
- Implement web applications using HTML, CSS, JavaScript, PHP, and MySQL
- Acknowledge common security vulnerabilities in web applications
- Mitigate against common security vulnerabilities in web applications

9.1.3 Mapping to NIST Nice Framework

This laboratory exercise is intended to increase the student's awareness of web application vulnerabilities. Additionally, the student will learn how to mitigate against these vulnerabilities. The student will have to perform small portions of programming in order to harden the vulnerable application. The student should have improved familiarity with HTML, MySQL, PHP, and Nginx upon successful completion of this exercise. This laboratory exercise maps to the following KSAs from the NIST NICE Framework:

- Cybersecurity and Privacy Principles (K0004)
- Cyber Threats and Vulnerabilities (K0005)
- Application Vulnerabilities (K0006)
- Data Administration (K0020)
- Database Management Systems (K0023)
- Programming Language Structures and Logic (K0068)
- Query Languages (K0069)
- Application Security Threats and Vulnerabilities (K0070)
- Secure Coding Techniques (K0140)
- Database Access Applicaiton Programming (K0197)
- Test and Evaluation Processes (K0250)
- Hacking Methodologies (K0310)
- Database Theory (K0420)
- Conducting Queries (S0013)
- Conducting Test Events (S0015)

- Designing Countermeasures to Identified Security Risks (S0022)
- Generate Queries (S0037)
- Writing Code (S0060)
- Recognize Types of Vulnerabilities (S0078)
- Applying Secure Coding Techniques (S0172)
- Develop Secure Software (A0047)
- Maintain Databases (A0176)

9.1.4 Necessary Background and Expected Completion Time

This laboratory exercise can be completed by students with varying background and experience. The following categories should help identify approximately how much time (in minutes) will be necessary to complete the laboratory exercise, for a student meeting the criteria for the respective experience level.

- Beginner: A student in this category has little to no experience with web application concepts or web development. This student should follow the walkthrough guide as well have instructor guidance.
- Intermediate: A student in this category is familiar with the concepts surrounding web applications and web development and may have some experience with web development. This student has little to no experience in web application hardening and security. This student should reference the walkthrough guide as needed.
- Advanced: A student in this category is experienced with web development and is familiar with web application hardening and security concepts. This student may need to reference the walkthrough to see how the vulnerable web application was implemented.

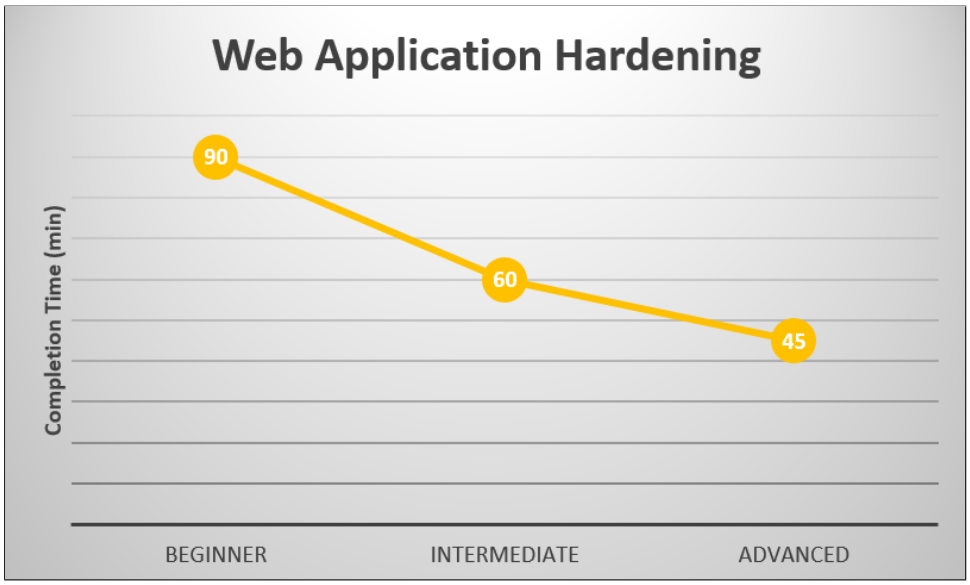

Figure 9.1: *Web Application Hardening Laboratory Exercise Expected Completion Time (min)*

9.1.5 Configuration and Setup

The machine used in this laboratory exercise is an installation of Ubuntu 16.04. The machine reaches out to a GitHub repo to clone files necessary to complete the exercises. Additionally, Nginx, PHP, and MySQL are installed. The initialization script creates and populates the MySQL databases and tables. Additionally, the script creates the vulnerable web application. The machine is configured using the initialization script, listing **9.1**.

Listing 9.1: *hw-initializationscript.sh*

```
#!/bin/bash

sudo dpkg --configure -a


#grab the github repo
sudo apt-get install git
sudo rm -r CYOTEE
sudo git clone https://github.com/CenterForSecureAndDependableSystems/CYOTEE.git

#install nginx
sudo apt-get install nginx -y

#install php
sudo apt-get install php-fpm php-mysql -y

#install MySQL
sudo apt-get install mysql-server -y

sudo cp CYOTEE/CYOTEE_Code/VulnerableCode/* /var/www/html/

sudo mv /var/www/html/default /etc/nginx/sites-available/default

sudo mysql -u root -e "DROP DATABASE test";
sudo mysql -u root -e "CREATE DATABASE test";

sudo mysql -u root -e "CREATE USER 'newuser'@'localhost' IDENTIFIED BY 'newpass'"
    ;
sudo mysql -u root -e "GRANT ALL PRIVILEGES ON test.* TO 'newuser'@'localhost'";

sudo mysql -u root test < CYOTEE/CYOTEE_Code/SQL/test.sql

sudo service nginx restart
```

9.1.6 Vulnerability Overview

1. SQL Injection
2. User Access not Controlled
3. Lack of Input Validation

9.1.7 Challenges

1. Implement and run the vulnerable web application
2. Identify where in the vulnerabilities exist
    - SQL Injection
    - Lack of Least Privilege Implementation

- Lack of Input Validation

3. Mitigate against vulnerabilities

   - Mitigate against SQL Injection
   - Implement a form of least privilege so that only necessary MySQL users have access to their respective pages
   - Perform input validation on the date field to ensure that a valid date is entered

9.2 Solutions and Guided Walkthrough

9.2.1 Solutions

The code for the hardened web application (after mitigating against the vulnerabilities) is available on the CYOTEE GitHub page at: *https://github.com/CenterForSecureAndDependableSystems/CYOTEE/*. You can copy the files from the hardened application to your local web server's /var/www/html directory. If you copy the files, discuss with your peers and/or colleagues how you would have approached the tasks if starting from scratch.

1. Implement and run the vulnerable web application

   Follow the walkthrough to set up the vulnerable application

2. Identify where in the vulnerabilities exist

   - SQL Injection

     This vulnerability exists in the PHP file in which a static MySQL command has user inputs placed in via the HTTP GET method

   - Lack of Least Privilege Implementation

     This vulnerability exists because the same database user accesses all tables in each PHP file

   - Lack of Input Validation

     This vulnerability exists because the input the user places into the start date field is not validated

3. Mitigate against vulnerabilities

    - Mitigate against SQL Injection

      The use of prepared statements in PHP can prevent against some forms of SQL injection.

    - Implement a form of least privilege so that only necessary MySQL users have access to their respective pages

      Create a new MySQL user for each table in the database which must be accessed. Grant each user access to their respective table. Set that user as the user connecting to the database in the PHP files.

    - Perform input validation on the date field to ensure that a valid date is entered

      Rather than just accepting any input from the user, ensure that the value is in date format ***MM/DD/YYYY*** *and that the date is question actually occurred.*

9.2.2 Guided Walkthrough

In order to complete the challenges in this laboratory exercise, see the steps in this walkthrough.

Least Privilege

Least privilege is a concept used daily, but seems to seldom be implemented in computing. At the simplest level, the principle of least privilege applies to an asset and a set of individuals who may access the asset. The principle of least privilege suggests that only individuals who absolutely require access to the asset should be authorized for that access. Consider the following: You and your family have moved into a new home. You make four copies of the house key, one for each member of the family. Upon more thinking, you come to the realization that your children are dropped off and picked up from school so they never enter or exit the house on their own and therefore, giving them a key is unnecessary. If the children were to lose the key, the security of home has been compromised. By recognizing that your children never needed their own keys, you have implemented a least privilege scheme in which only individuals who

require access to an asset (in this case a key) are granted that access. This is an example of implementing least privilege in the real world.

Web applications require multiple layers of security. One of the simplest layers of security to implement in web applications is least privilege; despite being simple, it frequently goes unimplemented. A typical web application will interact with a database via PHP. A simple way of implementing least privilege is to create multiple database users and grant each user different privileges based on what actions they must perform. An in depth guide of how to perform these tasks will be outlined in the **Implementation** section [36].

SQL Injection

Many web applications include some sort text entry field to allow the user to enter data such as a username and/or password among other possibilities. The value entered into the text field is then typically used in some sort of a structured query language (SQL) query [37]. SQL is used to manage and query relational databases in computing. Queries allow a user to read data from, insert data into, and delete data from a database, among other actions. Database management applications such as MySQL [38] have specific syntax for queries. A simple web application with little security implemented will typically insert the user entered value directly into a partially constructed MySQL query. For example, a sample MySQL query in which the user entered value is represented by the variable `$userval` may look something like:

```
SELECT * FROM table WHERE user='$userval';
```

In this case, the query, when executed, will allow the user to view all data fields for a row in which the **user** field has the value indicated by the user input. When using partially constructed queries into which the user's value is dropped in, the SQL Injection [39] vulnerability is introduced.

A malicious user can take advantage of the fact that the web application takes the user input as is and enters it into the partially constructed query. If the malicious would like to be able to view all rows of data from the table, he/she can enter a specifically formatted value into the text input field to achieve the task. An example of this input is:

```
'  OR '1=1
```

Although this input looks like something out of a poorly written math textbook, let us identify what it does to the query. By inserting that value into the partially constructed query, the full query is:

```
SELECT * FROM table WHERE user='' OR '1=1';
```

Analyzing this query, the malicious intent is visible. The malicious user has formed a query such that MySQL will return a row if either the name field is blank, ***or*** if 1=1 (this always evaluates to true). Because 1=1 always evaluates to true, MySQL will return every row in the table. Other values can be entered into the input field to perform other SQL queries.

SQL Injection Mitigation

Given that allowing a user to input a value through a text field introduces a vulnerability, the obvious question arises: "What do I do if I need to let the user enter a value?" One suggestion may be to check the value that the user entered and analyze it for potential SQL Injection; upon trying to perform this task, one finds it to be quite futile as there are an infinite number of ways a malicious user could format their input to perform a malicious task. Rather, it is best to handle the actual issue: the user input is inserted into the query exactly as it is read in. Rather than allowing the user input to be placed into the partially constructed query, it would be useful to have everything the user enters interpreted as a string literal. This can be accomplished through the use of what are known as *prepared statements* [40].

Prepared statements can be implemented through PHP to allow your web application to interact with the MySQL database. A prepared statement is implemented in three stages:

1. Prepare
2. Bind
3. Execute

In the preparation stage, the user writes a SQL statement template with certain values left unfilled. These values are specified by question marks (?) and are called parameters. Using the same query as discussed in the SQL Injection section, a sample PHP line to prepare a statement would be:

```
$query = $conn->prepare("SELECT * FROM table WHERE name=?");
```

In the bind stage, the user specifies what values should be used to fill the previously ambiguous fields identified by question marks. In addition to specifying the value, the user must also specify the type of the value. A sample PHP line to bind parameters would be:

```
$query->bind_param("s",$userval);
```

In the execution phase, the user simply specifies that he/she would like to execute the query which has been prepared and bound. A sample PHP line to execute the SQL query would be:

```
$query->execute();
```

Implementing prepared statements over traditional dynamic MySQL queries in PHP is not a challenging but improves the security of the web application significantly. Because the user's input value is being interpreted literally and there are no quotes to escape properly in the prepared statement as there were in the traditional statement, a malicious user cannot carry out the same SQL Injection described previously.

User Input Validation

Web applications often allow a user to enter data into a field which must be of a certain format, for example a birthday. However, web applications often do not validate that the value entered was actually in the format required. Consider the following application: employees at a company must enter the date they started working for the company. In the event that the desired input format is **MM/DD/YYYY**, what are some possible ways in which invalid data could be entered? A few possibilities which commonly strike a developer are that maybe the user enters the date in the wrong format such as **DD/MM/YYYY** or **MM/DD/YY**. Another common mistake could be that the user simply does not enter a date and instead provides some sort of bogus input.

Provided that the user did actually enter a date in the correct format, another check which should be performed is whether the date entered is a valid date. Examples of invalid dates include **11/31/2019** and **02/29/2019**. The first of the two invalid dates is invalid because November never has 31 days in the Gregorian calendar. The latter of the two is invalid because 2019 was not a leap year, therefore meaning February 29, 2019 never occurred.

Implementation

In order to demonstrate the steps for taking a vulnerable, basic web application and implementing the aforementioned security measures, a series of challenges has been created. To complete these challenges, an individual will have to successfully implement each of the security measures and therefore create a more secure web application. A detailed walkthrough of how this exercise was developed, as well as how you can recreate it is outlined in the following sections.

Implementing Least Privilege

Note that in the vulnerable web application, the PHP pages which establish communications with the MySQL database have the root user being used to logon to MySQL. The root user is granted what is known as **ALL PRIVILEGES** in MySQL; this means the root user has the ability to perform any action, on all tables in any database, in MySQL. Inspecting the PHP pages will reveal that one of the pages performs a query using the **SELECT** command and the other performs a query using the **INSERT** command.

This allows a secure programmer to implement least privilege. There is no reason to have the root user logging into the database from the PHP page, rather create a new user. Moreover, that new user only needs to have SELECT privileges on the table referenced in the code (in this case, the **employees** table in the **test** database. Additionally, another user can be created and granted only INSERT privileges on the **startdates** table in the **test** database.

The first step towards creating the new users is to log into MySQL using the root user. As a reminder, this can be done by entering the following command in the terminal and entering the root user password:

```
$ sudo mysql -u root -p
```

The next step is to create the two new users; in order to do so, enter the following command in the MySQL console:

```
mysql> CREATE USER '<webapp1>'@'localhost' IDENTIFIED BY "<webapp1>";
mysql> CREATE USER '<webapp2>'@'localhost' IDENTIFIED BY "<webapp2>";
```

You can replace the usernames and passwords (indicated by blue text) with whatever you

Listing 9.2: *Vulnerable Authorization Query*

```
$sql = "SELECT * FROM employees WHERE name='" . $_GET["user"] . "' AND password='
    " . $_GET["pass"] . "';";

$result = $conn->query($sql);
```

Listing 9.3: *Hardened Authorization Query*

```
$sql = $conn->prepare("SELECT * FROM employees WHERE name=? AND password=?");

$sql->bind_param("ss",$_GET['user'],$_GET['pass']);

$sql->execute();
```

would like but those are the values which are used in the final code.

Now that two new users have been created, it is necessary to grant the appropriate privileges to each user. Currently the users only have **USE** privileges. In order to grant privileges, enter the following into the MySQL console:

```
mysql> GRANT SELECT ON test.employees TO 'webapp1'@'localhost';
mysql> GRANT INSERT ON test.startdates TO 'webapp2'@'localhost';
```

This will allow the **webapp1** user to read data from only the **employees** table and the **webapp2** user to insert data into only the **startdates** table, both in the **test** database.

Utilizing Prepared Statements

The use of prepared statements is one which allows a web application developer to mitigate against SQL Injection. As discussed previously in this document, prepared statements take the user input and insert them as string literals into the prepared statement rather than inserting the variable value directly into the dynamic SQL query template. In the web application, there are two files which require the use of user input in a SQL query; both of these files should implement prepared statements. In the file *vw-auth.php*, the original (vulnerable) lines associated with the query can be seen in listing **9.2**.

After implementing the prepared statements in the **hw-auth.php** file, the lines associated with the query can be seen in listing **9.3**.

In the file *vw-insert.php*, the original (vulnerable) lines associated with the query can be seen

Listing 9.4: *Vulnerable Insertion Query*

```
$sql = "INSERT INTO startdates (name, date) VALUES (\"$user\",\"$date\");";

$conn->query($sql);
```

Listing 9.5: *Hardened Insertion Query*

```
$sql = $conn->prepare("INSERT INTO startdates (name,date) VALUES (?,?)");

$sql->bind_param("ss",$user,$date);

$sql->execute();
```

in listing **9.4**.

After implementing the prepared statements in the **hw-insert.php**, the lines associated with the query can be seen in listing **9.5**.

Viewing the differences between the vulnerable code and the hardened code will reveal that there is not much extra work required to implement the prepared statement. Due to the prepared statement formatting the string as a literal, the web application developer does not need to ensure that the double and single quotes are placed and escaped appropriately in the SQL query. In making the change from the vulnerable code to the hardened code using prepared statements, the process for returning the result of the query and printing is also changed slightly.

Performing Input Validation

Input validation may very well be a programmer's dream and nightmare simultaneously. A dream because it improves the security of the application but a nightmare because of the amount of effort needed to perform input validation. It can be a tedious task, particularly considering the potential edge cases for what types of input could be considered invalid. In this web application, input validation should be performed when the end-user is tasked with entering a date on the *hw-insert.php* page. At this point, prepared statements have been implemented in this file but nothing has been done to ensure that the user entered a valid date. As noted previously in this document, there are a few checks which should be performed in relation to input validation of the input provided on the *hw-insert.php* page.

The page requests that the user insert a date in **MM/DD/YYYY** format. The first check

Listing 9.6: *Date Format Matching Regex*

```
1 $regex = "/(\d{2})\/(\d{2})\/(\d{4})/";
2
3 preg_match($regex,$date,$matches);
```

which should be performed is whether or not the input is even in this format. This task can be accomplished using regular expression (regex) matching. The lines of code seen in listing **9.6**, in the hardened *hw-insert.php*, file accomplish this task.

The code seen in listing **9.6** simply checks if the input provided by the user (held in the `$date` variable) matches the regex. The regex will match any input which is in the format of two digits, followed by a forward slash, followed by two digits, followed by a forward slash, followed by four digits (digits can be any number between 0 and 9).

After determining that the input entered is in the correct **MM/DD/YYYY** format, one should begin checking for other invalid inputs such as invalid dates. Examples of invalid dates include the following:

- If the month portion is less than 1 or more than 12.
- If the day portion is less than 1 or more than 31.
- If the month portion is April, June, September, or November and the day portion has more than 30.
- If the month portion is February and the day portion is more than 29.
- If the month portion is February and the day portion is 29 and the year portion is not a leap year.

These checks are easily performed in PHP but first some overhead is required. The date field is entered in as a whole in **MM/DD/YYYY** format but in order to perform the inequality checks listed above, the portions of the date must be split into individual parts. This is done by splitting the string on a delimiter character. The natural choice for the delimiter character in this case is the forward slash (**/**). This is done in PHP by using the **explode** function. The function returns an array of the substrings obtained by splitting on the delimiter character. The

Listing 9.7: *Split Date into Substrings*

```
$da = explode("/",$date);
$m = intval($da[0]);
$d = intval($da[1]);
$y = intval($da[2]);
```

Listing 9.8: *Validate Month Value*

```
if($m < 1 || $m > 12)
{
        header("Location: hw-invaliddate.html");
        exit();
}
```

code seen in listing **9.7**, performs the splitting and stores the individual parts of the date in individual variables.

With the individual parts of the date stored in individual variables, inequality checks can be performed. Refer to the list below for the code snippets:

- If the month portion is less than 1 or more than 12, code seen in listing **9.8**.
- If the day portion is less than 1 or more than 31, code seen in listing **9.9**.
- If the month portion is April, June, September, or November and the day portion has more than 30, code seen in listing **9.10**.
- If the month portion is February and the day portion is more than 29, code seen in listing **9.11**.
- If the month portion is February and the day portion is 29 and the year portion is not a leap year, code seen in listing **9.12**.

Listing 9.9: *Validate Day in All Months*

```
if($d < 1 || $d > 31)
{
        header("Location: hw-invaliddate.html");
        exit();
}
```

Listing 9.10: *Validate Day in 30 Day Months*

```
if($m == 4 || $m == 6 || $m == 9 || $m == 11)
{
        if($d == 31)
        {
                header("Location: invaliddate.html");
                exit();
        }
}
```

Listing 9.11: *Validate Day in February for Any Year*

```
if($m == 2)
{
        if($d > 29)
        {
                header("Location: invaliddate.html");
                exit();
        }
        ...

}
```

Listing 9.12: *Validate Day in February for Leap Years*

```
if($m == 2)
{
        ...

        if($d == 29)
        {
                if( (($y % 4 == 0) && ($y % 100 != 0)) || ($y % 400 == 0) )
                {
                        //this is a leap year, which is valid
                }
                else
                {
                        header("Location: invaliddate.html");
                        exit();
                }
        }
}
```

Hardened Code

The hardened code, file by file, can be seen in the following listings, along with their respective file names.

Listing 9.13: *hw-index.html*

```
<html>
	<body style="background-color: black; color: white">
		<div style="text-align: center">
			<form action="auth.php" method="get">
				Username: <input type="text" name="user"><br>
				Password: <input type="text" name="pass"><br>
				<input type="submit">
			</form>
		</div>
	</body>
</html>
```

Listing 9.14: *hw-auth.php*

```php
<?php

$servername = "localhost";
$username = "webapp1";
$password = "webapp1";
$dbname = "test";

$conn = new mysqli($servername,$username,$password,$dbname);

$sql = $conn->prepare("SELECT * FROM employees WHERE name=? AND password=?");

$sql->bind_param("ss",$_GET['user'],$_GET['pass']);

$sql->execute();

$result = $sql->get_result();

if($result->num_rows > 0)
{
        print "Click you name to access your date entry form!<br>";
        while($row = $result->fetch_assoc())
        {
                $n = $row['name'];
                print "<a href='insertdate.php?name=$n'>" . $n . "</a><br><br>";

        }
}
else
{
        header("Location: invalid.html");
}

$conn->close();
?>
```

Listing 9.15: *hw-insertdate.php*

```php
<?php
        $n = $_GET['name'];
?>

<html>
        <body style="background-color: black; color: white">
                <div style="text-align: center">
                        <form action="insert.php" method="get">
                                Enter a date in MM/DD/YYYY format<br>
                                <input type="text" name="date"><br>
                                <input type="text" name="user" value="<?php echo $n;
                                    ?>" readonly><br>
                                <input type="submit">
                        </form>
                </div>
        </body>
</html>
```

Listing 9.16: *hw-insert.php*

```php
<?php

$servername = "localhost";
$username = "webapp2";
$password = "webapp2";
$dbname = "test";

$user = $_GET['user'];
$date = $_GET['date'];

$regex = "/(\d{2})\/(\d{2})\/(\d{4})/";
preg_match($regex,$date,$matches);

if($matches == NULL)
{
	header("Location: invaliddate.html");
	exit();
}
else
{
	$da = explode("/",$date);
	$m = intval($da[0]);
	$d = intval($da[1]);
	$y = intval($da[2]);

	if($m < 1 || $m > 12)
	{
		header("Location: invaliddate.html");
		exit();
	}

	if($d < 1 || $d > 31)
	{
		header("Location: invaliddate.html");
		exit();
	}

	if($m == 4 || $m == 6 || $m == 9 || $m == 11)
	{
		if($d == 31)
		{
			header("Location: invaliddate.html");
			exit();
		}
	}

	if($m == 2)
	{
		if($d > 29)
		{
			header("Location: invaliddate.html");
			exit();
		}

		if($d == 29)
		{
			if( (($y % 4 == 0) && ($y % 100 != 0)) || ($y % 400 == 0) )
			{
				//this is a leap year, which is valid
```

```
                        }
                        else
                        {
                                header("Location: invaliddate.html");
                                exit();
                        }
                }
        }
}


$conn = new mysqli($servername,$username,$password,$dbname);

if($conn->connect_error)
{
        die("Connection Failed: " . $conn->connect_error);
}

$sql = $conn->prepare("INSERT INTO startdates (name,date) VALUES (?,?)");
$sql->bind_param("ss",$user,$date);
$sql->execute();

?>

<html>
        <body style="background-color: black; color: white">
                <div style="text-align: center">
                        <h1>Insert Complete!</h1>
                        <button onclick="location.href='index.html'">Return to Login
                             Page</button>
                </div>
        </body>
</html>
```

Listing 9.17: *hw-invaliddate.html*

```
<html>
        <body style="background-color: black; color: white">
                <div style="text-align: center">
                        <h1>Invalid Date Entered!</h1>
                        <button onclick="location.href='index.html'">Return to Login
                             Page</button>
                </div>
        </body>
</html>
```

Listing 9.18: *hw-invalid.html*

```
<html>
        <body style="background-color: black; color: white">
                <div style="text-align: center">
                        <h1>Invalid Logon Attempt!</h1>
                        <button onclick="location.href='index.html'">Return to Login
                             Page</button>
                </div>
        </body>
</html>
```

Chapter 10: Active Directory Usage & Hardening

In this laboratory exercise, the student will be introduced to the basic concepts of Active Directory and configure a domain, create group policy objects, and harden a domain controller.

10.1 Laboratory Exercise

Active Directory Usage & Hardening

10.1.1 Specifications

This lab exercise will be performed on Windows Server 2016, hereon referred to as **the Windows Server**. A trial version of Windows Server 2016 was used for development.

10.1.2 Learning Objectives

- Configure Windows Server
- Configure Active Directory
- Understand the Concept of Active Directory and Domain Controllers

10.1.3 Mapping to NIST Nice Framework

This laboratory exercise is intended to increase the student's familiarity with Active Directory and Group Policy. This exercise is skills application heavy. This laboratory exercise maps to the following KSAs from the NIST NICE Framework:

- Cybersecurity and Privacy Principles (K0004)
- Cyber Threats and Vulnerabilities (K0005)
- Test and Evaluation Processes (K0250)
- Recognize Types of Vulnerabilities (S0078)
- Apply Cybersecurity and Privacy Principles to Organizational Requirements (S0367)
- Apply Cybersecurity and Privacy Principles to Organizational Requirements (A0123)

10.1.4 Necessary Background and Expected Completion Time

This laboratory exercise can be completed by students with varying background and experience. The following categories should help identify approximately how much time (in minutes) will be necessary to complete the laboratory exercise, for a student meeting the criteria for the respective experience level.

- Beginner: A student in this category has little to no experience with the basics of active directory, group policy, and Windows Server.
- Intermediate: A student in this category is familiar with the basics of active directory, group policy, and Windows Server. This student may have hands-on experience using Windows Server.
- Advanced: A student in this category has experience with the basics of active directory, group policy, and Windows Server. This student also has hands-on experience using Windows Server, possibly a background as a systems administrator.

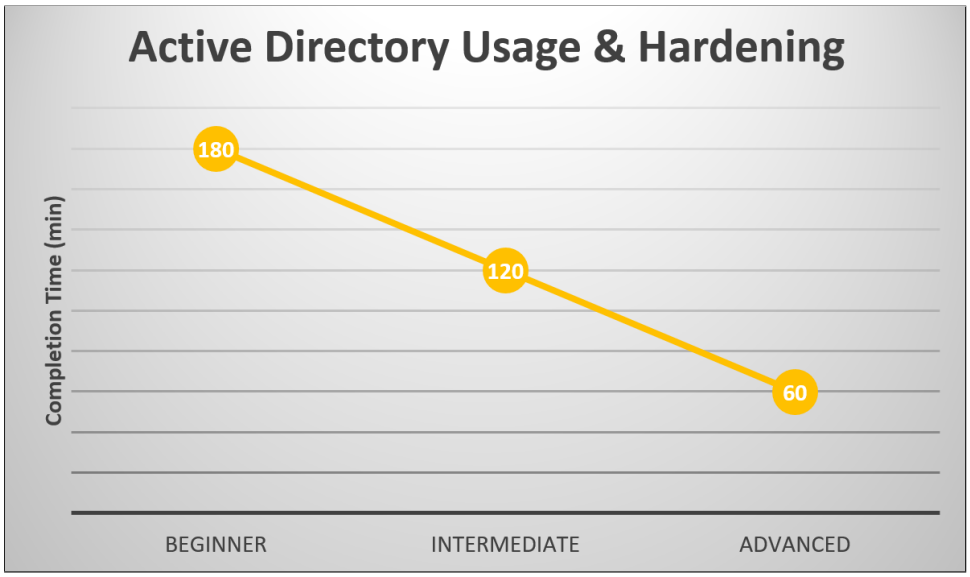

Figure 10.1: *Active Directory Usage & Hardening Laboratory Exercise Expected Completion Time (min)*

10.1.5 Configuration and Setup

The machine for this laboratory exercise does not require any custom configuration as the exercise requires the student to configure a Windows Server using Active Directory and Group Policies.

10.1.6 Challenges

Brief Introduction to Active Directory

Active Directory is a service typically found on Windows Server operating systems. Active Directory is a suite of configuration tools created by Microsoft which can be used to perform remote administration of systems [93]. Active Directory, short for the term *Active Directory Domain Services* [93], is a service which can be enabled on a Windows Server; when used to control and configure a group of machines, this server becomes known as a *Domain Controller* [93]. A domain controller can be used to remotely push policies, known as *Group Policies*, to all systems on the domain. In the following tasks, you will enable and configure Active Directory on a Windows Server, create Group Policy Objects, and perform a few basic domain controller hardening tasks.

1. Configure the Windows Server to have the hostname `CYOTEE-DC`
2. Configure the Windows Server to have an IPv4 address of `192.168.7.20` and a default gateway of `192.168.7.1`.
3. Configure the Windows Server to use itself as its preferred DNS Server and do not set an alternate DNS server.
4. Configure the Windows Server to reflect your time zone.
5. Install Active Directory on the Windows Server.
6. Create a new domain `cyotee.local` in a new forest.
7. Configure the Forest Functional Level to Windows Server 2008.
8. Configure the Domain Functional Level to Windows Server 2008.
9. Ensure that the Windows Server is configured as a Global Catalog server.
10. Ensure that the Windows Server is configured as a DNS Server.
11. Configure DNS on the Windows Server to use the external DNS server with IPv4 address `192.168.7.30` as a forwarder.
12. Configure the Windows Server to not allow **root hints**.
13. Create the following Organizational Units in Active Directory.
    - Professors
    - Staff
14. Create the following three users in each of the Organizational Units. The username naming convention should be <firstname>.<lastname>@<domain>. Spaces or punctuation within a name should be omitted. Create a default password for each user (`Password123`) but require they change their password at the next login.
    - Professors

  - Emma Castillo
  - Jeanette Wise
  - Bernadette Rivera
- Staff
  - Boyd Harmon
  - Jeremiah Houston
  - Adrian Miles

15. Create a new Security Group in each Organizational Unit with the same name as the Organizational Unit. The Security Group should be Global.

16. Add the three users in each Organizational Unit to the respective Security Group.

17. Create a Shared Folder for each Organizational Unit. The name of the Shared Folder should be the same as the name of the Organizational Unit with the word **Folder** appended to it (ex: StaffFolder). The only users who belong to the Organizational Unit should have access to the folder. The network path to the folder should be `\server\share\<foldername>`.

18. Create a new site called CYOTEE. Add the Windows Server to the new site.

19. Create a new Group Policy Object (GPO) and link it to the domain. The new GPO should match the following specification:

    - Name the GPO `cyotee-pol`.
    - Create a Folder Redirection policy to map the `Documents` folder to a network share on the Windows Server located at `\\server\share\network-share`.
    - Remove `Run` from the Start Menu.
    - Configure Control Panel to only start in icon view.
    - Block use of `regedit.exe` to mitigate against users editing the registry.
    - Disable the ability for users to access the command prompt.
    - Disable the ability for users to change the system time.

- Block use of `taskmgr.exe` to disable the ability for users to access the Task Manager.

20. Install the File Server Resource Manager role. Configure it according to the following specification:

    - Install the File Server Resource Manager on the Windows Server.
    - Create a text file in the user's `Desktop` directory containing the word *classified*.
    - Create a Classification Property called `Classified Property`.
    - Create a Classification Rule called **Classified Files** for files which contain the word *classified*.
    - Apply the Classified Files rule to the user's Desktop directory.
    - Verify whether or not the classification rule was applied properly by running the classification.

21. Create a File Screen which meets the following specifications:

    - Block all `.bat` and `.exe` files from running.
    - Apply the screen to the `My Documents` directory.
    - Generate an appropriate warning message.
    - Verify that the screen works by attempting to run a .bat file in the affected directory.

22. Reduce the Attack Surface by Removing Browsers

    One suggestion made in Microsoft's online documentation of Windows Server, in order to improve the security of domain controllers, is to remove all web browsers [94]. The article states:

> *"Browsing the Internet (or an infected intranet) from one of the most powerful computers in a Windows infrastructure using a highly privileged account (which are the only accounts permitted to log on locally to domain controllers by default) presents an extraordinary risk to an organization's security. Whether via a drive by download or by download of malware-infected "utilities," attackers can gain access to everything they need to completely compromise or destroy the Active Directory environment."*

Task: To mitigate against this risk, remove all web browsers from the domain controller (Windows Server).

23. Apply Principles of Least Privilege

    In Active Directory, there are three levels of administrators, namely: Built-In Admins (BA), Domain Admins (DA), and Enterprise Admins (EA). The BA group tends to have many users because they are thought to have less privileges than those in the DA and EA groups. This may be true, the base privileges granted to BA members is less than that of DA and EA members. This becomes irrelevant when noting that a member of any of the three groups can modify the membership of other groups, effectively gaining administrative control over all systems in the nested group [95]. It is suggested that administrative privileges only be granted to users who absolutely require the role to perform their tasking [96].

    Task: Reduce the number of administrators on a domain, including built-in administrators. Regularly monitor the administrators, as attackers will create domain admin accounts to gain control over a domain.

24. Hardening the Administrator Accounts

    Administrator accounts have increased privileges and can often be used to wreak havoc on a domain if compromised. Some suggestions for securing the administrator accounts include dual factor authentication and anti-delegation [95].

    Task 1: Require that administrator accounts need a smart card for logon.

    Task 2: Mark the administrator account as sensitive and cannot be delegated.

10.2 Solutions and Guided Walkthrough

Walkthrough

In order to complete the challenges in this laboratory exercise, see the steps in this walkthrough.

1. Navigate to `Start Menu` → `Control Panel` → `System` → `Computer Name, Domain, and Workgroup Settings` → `Change Settings` and then click Change to rename the computer.

2. Navigate to `Start Menu` → `Control Panel` → `Network and Sharing Center` → `Change Adapter Settings` → `Right-Click on the Local Area Connection and select Properties` → `Select Internet Protocol Version 4` → `Select Properties`. From here, configure the desired IP address settings.

3. Follow the steps in Step 2 and configure the desired DNS settings.

4. Navigate to `Start Menu` → `Control Panel` → `Date and Time` → `Change Time Zone`. From here, select the desired time zone settings.

5. In the Server Manager, right click on `Roles` and select `Add Roles`. Follow the steps in the wizard and select `Active Directory Domain Services` as the role to install.

6. During the Active Directory installation process, you should be asked to `Create a New Domain`. Select `Domain in a New Forest` and then click Next. Enter the name of the new domain.

7. During the Active Directory installation process, you should configure the Forest functional level to the desired setting when prompted.

8. During the Active Directory installation process, you should configure the Domain functional level to the desired setting when prompted.

9. During the Active Directory installation process, you should configure the Windows Server as a Global Catalog Server by checking the box next to Global Catalog Server. It may already be checked by default.

10. During the Active Directory installation process, you should configure the Windows Server as a DNS Server by checking the box next to DNS Server. This box is typically not checked by default, ensure this is checked.

11. In the Server Manager, expand the `Roles` tab. Expand the `DNS Server` tab. Expand the `DNS` tab. Then right-click the name of your server and select properties. Select the `Forwarders` tab. Click `Edit` and add the desired Forwarder IP address.

12. In the `Forwarders` tab mentioned in Step 11, uncheck the box labeled `Use root hints if no forwarders are available`.

13. In the Server Manager, expand the `Roles` tab. Expand the `Active Directory Domain Services` tab. Expand the `Active Directory Users and Computers` tab. Then right-click the name of your domain and select New. Select Organizational Unit. Configure the names as desired.

14. Follow the directions in Step 13 to navigate to your domain. Next, expand your domain, you should see the Organizational Units which were created in Step 13. Right-click on the Organizational Unit and select New. Next, select User. Follow the specifications to create the users appropriately.

15. Follow the directions in Step 14 to navigate to your Organizational Units. Right-click on the Organizational Unit and select New. Next, select Group. Name the group according to the specification. Ensure that the `Group Scope` and `Group Type` are configured according to the specification as well.

16. Follow the directions in Step 14 to navigate to your Organizational Units. Next select all of the users which need to be added to the Security Group. When all are selected, right-click on one of them and select `Add to Group`. Type in the name of the group.

17. Follow the directions in Step 14 to navigate to your Organizational Units. Next, right-click the Organizational Unit and select New. Next, select Shared Folder. Configure the name and network path according to the specification.

18. In the Server Manager, expand the `Roles` tab. Expand the `Active Directory Domain Services` tab. Expand the `Active Directory Sites and Services` tab. Right-click the `Sites` tab. Select New and then select `Site`. To add the server to the site, right-click the site and select New. Next, select Server. Enter the name of your server.

19. In the Server Manager, expand the `Features` tab. Expand the `Forest` tab. Expand the `Domains` tab. Expand the tab with your domain name. Right-click the `Domain Controllers` tab. Select `Create a GPO in this domain, and Link it here`.

    - Name the GPO according to the specification when prompted. *If not immediately brought to the Group Policy Management Editor, go back to the Domain Controllers tab and right-click on the name of your new GPO and select Edit.*
    - In the Group Policy Management Editor, navigate to `User Configuration` → `Policies` → `Windows Settings` → `Folder Redirection` and configure according to the specification.
    - In the Group Policy Management Editor, navigate to `User Configuration` → `Policies` → `Administrative Templates` → `Start Menu and Taskbar` and then find the setting titled **Remove Run menu from Start Menu**. Enable that setting.
    - In the Group Policy Management Editor, navigate to `User Configuration` → `Policies` → `Administrative Templates` → `Control Panel` and find the setting titled **Always Open all Control Panel Items when Opening Control Panel**. Enable that setting.
    - In the Group Policy Management Editor, navigate to `User Configuration` → `Policies` → `Administrative Templates` → `System` and then find setting titled **Prevent Access to Registry Editing Tools**. Enable that setting.
    - In the Group Policy Management Editor, navigate to `User Configuration` →

`Policies` → `Administrative Templates` → `System` and then find setting titled **Prevent Access to the command prompt**. Enable that setting.

- In the Group Policy Management Editor, navigate to `User Configuration` → `Policies` → `Administrative Templates` → `System` → `Locale Services` and then find setting titled **Disallow User Override of Locale Settings**. Enable that setting.
- In the Group Policy Management Editor, navigate to `User Configuration` → `Policies` → `Administrative Templates` → `System` → `Ctrl + Alt + Del Options` and then find setting titled **Remove Task Manager**. Enable that setting.

20. From the Server Manager, follow these steps:

    - Right-click on the `Roles` tab. Select `Add Roles`. Navigate through the Wizard. Ensure that `File Services` is selected to be installed.
    - Navigate to `Roles` → `File Services` → `Share and Storage Management` → `File Server Resource Manager` → `Classification Management`. Right-click on `Classification Properties` and select `Create Property`. Give the property a name and set the type as Yes/No. Next, right-click `Classification Rules` and select `Create a New Rule`. Name the rule and select the scope to which it should apply (which folders it should be applied to) based on the specification. Then select the `Classification` tab. Set the `Classification Mechanism` to `Content Classifier`. Set the `Property Name` to the name of the property you created earlier in this step. Set the property value accordingly. Next, select Advanced. Click on the `Additional Classification Parameters` tab. Set the Name field to **String** and set the value according to the specification.
    - Run the rule by right-clicking the `Classification Rules` tab and selecting `Run Classification with all Rules Now`.

21. To create a File Screen, follow these steps:

    - Navigate to `Roles` → `Services` → `Share and Storage Management` →

`File Server Resource Manager` → `File Screening Management` and right-click on `File Screens`.

- Select `Create File Screen`.
- Specify the path to the directory on which the screen should apply.
- Either use an existing file screen or create a custom screen.

22. To remove a browser on Windows Server, open the **Control Panel**. From there, navigate to Programs → Programs and Features → Uninstall or Change a Program. From here uninstall any browsers installed on the machine.

23. Apply Principles of Least Privilege

    Check your users and user groups to determine if there are users with unnecessary privileges. You can check this in the Server Manager. Navigate to **Roles** → **Active Directory Domain Services** → **Active Directory Users and Computers** → *Domain Name* → **Users**. Here, click the **Type** button in the top bar to sort by type, this will make it easier to see users vs groups. Ensure that the Guest account is disabled, this account is usually unprotected and unnecessary but provides an attack vector. Next, look through the remaining users, determine if they are necessary, if not, disable the account. Next, check the various security group and see what members are in them. To do this, right click the desired group, select **Properties**, then click on the **Members** tab. Ensure that a user is only in the group if necessary. One concept of least privilege is to only grant access as needed. Some individuals within an organization may need to perform domain services on occasion. Rather than leave that user account in an administrators group, only add them when necessary, and remove after. This ensures that accounts are not being compromised and used to perform privileged actions.

24. Hardening the Administrator Accounts

    Task 1: Require that administrator accounts need a smart card for logon.

    Task 2: Mark the administrator account as sensitive and cannot be delegated.

    For each of these tasks, access a user's properties by navigating to **Roles** → **Active**

Directory Domain Services → **Active Directory Users and Computers** → *Domain Name* → **Users** within the Server Manager. From here, right click on the desired user and select **Properties**. In the **Account** tab, there will be a sections titled **Account Options**. Scroll through the options and select the desired security features. *Note: In order to require the smart card login, your organization will need to have a public key infrastructure in place.*

Chapter 11: Performing Customer Service at Competitions

11.1 Common Customer Service Tasks at Competitions

This section discusses, in detail, common tasks required of the blue team with respect to customer service at cyber defense competitions.

11.1.1 Setting Up an Appropriate Voicemail Greeting

As technology advances, organizations are moving towards newer technological methods, encryption enabled services, VPNs, telecommuting, the Cloud, etc. One technological item which is still a staple in almost every organization is a telephone. It is important to ensure that an organization is appropriately staffed such that the telephone is answered when calls arrive. That said, even the best organization cannot always ensure that every call is answered. In the case that a staff member is not able to field a telephone call, it is integral to the continued success of the organization that an appropriate voicemail greeting is in place.

There are many components to an effective voicemail greeting. The first component to be addressed is the actual language and tone of voice used. Jackie Silver, a voice artist who has been in the industry for over 25 years, states that:

> *"Voice is the first connection a client has to the business - make it count!"* [68]

It can be convenient to use some sort of text to speech software or an automated voicemail system to record a greeting, but this can lead to the perception that an organization is not personal with their clients. In a world which is shifting towards automation, remaining personal with clients can be the leg up that an organization has over its competitors. A more personal greeting can also improve the overall mood of the caller, thus reducing the likelihood that a customer will be difficult over the phone. Silver states:

> *"Using a warm, relateable person for a business voice over is preferable to the automation-soundin, monotone voice that many businesses choose."* [68]

By ensuring that the voicemail greeting is personal, genuine, and warm, an organization sets a positive first impression with the client.

The next component of a quality voicemail greeting is brevity. If a client has gone the route of calling you on the telephone, they are likely hoping for a quick response rather than a delayed response via email. Ensure that your voicemail greeting does not leave the client waiting for an extended period of time before they can record their message. In an article from Statup Stockpile, author Ryan Bozeman suggests that an organization should aim for a voicemail greeting last 20 seconds. Less than 10 seconds likely means that the speaker was speaking to fast or all relevant information was not relayed. He states that a voicemail greeting should never exceed 30 seconds [69].

The next component of an effective voicemail greeting is ensuring that all pertinent information is relayed to caller. Keep the voicemail greeting short and to the point, but do not leave out necessary details. A successful voicemail greeting should include the following items:

- The name of the individual or organization the caller has reached.
- Express your apology for lack of availability.
- Normal business operating hours.
- Alternative methods of contact (if applicable).
- Request a message including the callers name (and organization if applicable), a call back number, and the reason for the call.
- A salutation.

An example of a high quality effective voicemail greeting script is shown below:

> *Thank you for calling **XYZ Corp**. We apologize in advance for missing your call. Our business operating hours are **8 AM to 5 PM Pacific Time, Monday to Friday**. You may also contact us at **customerservice@xyzcorp.com**. Please leave your name, telephone number, and your reason for calling today, and we will return your call as soon as possible. Please record your message at the tone and have a wonderful day.*

More examples of effective voicemail greetings are available in Silver's article [68], Bozeman's article [69], and an article from OnSIP [67].

11.1.2 Performing Effective Customer Service

In order for an organization to achieve lasting success, it is important that customers and clients have the ability to have their questions answered. This leads to a necessity for a customer service center of some sort. This does not inherently need to be a full team of 50 individuals, but can even just be a single individual who is on hand to assist customers and clients with their needs. Digital.gov, a product of the U.S. General Services Administration notes that customer contact centers serve as an effective liaison between the customers and the organization [63].

Effective customer service starts with establishing standards which must be met when interacting with customers. Concrete standards allow each individual in the organization to have an outline of what items must be met when interacting with a customer and allow individuals to hold them self and others accountable. Example standards are listed as part of the U.S. Department of Health & Human Services' Agency for Healthcare Research and Quality's Strategy 6Q Standards for Customer Service [64].

The U.S. Office of Personnel Management lists in a 1997 report that effective customer service is achieved by incorporating the following items into an organization [65]:

- **Goal Setting and Measurement**: Set goals and measure whether they were achieved.

Without a quantifiable or verifiable method of assessing goals, it is challenging to identify areas of strength and weakness.

- **Goal Setting and Feedback**: Having concrete feedback helps lead an organization to reevaluate its goals and set new goals which have more positive feedback.
- **Measurement and Feedback**: It is important that the measurement system is not shared with the same individuals who will be providing feedback. This can lead to skewed feedback.
- **Measurement and Rewards**: For individuals to participate in any exercise with real effort, there must be some sort of reward for participating in a genuine, honest manner.
- **Feedback and Rewards**: An organization must recognize individuals who have solicited positive feedback from customers. This leads to positive reinforcement for the individual as well as setting a standard for others to follow.

This process is exemplified in the following statement by Connellan and Zemke [66]:

> *"... if you set both a standard and a goal; if you involve indiviuals or teams in setting their targets; if you empower individuals and teams to make decisions on their own; if you combine goal setting with measurement of customer satisfaction tracked back to both individuals and teams; if you add positive coaching; if you celebrate progress; and if you use regular positive reinforcement for the right set of behaviors, then goal setting is a powerful, positive tool for sustaining Knock Your Socks Off Service."*

To review, in order to provide effective customer support and service, an organization needs to identify what the needs of the customers are. Next, the organization must set goals which aim to satisfy the customer needs. Using some form of measurement, customer satisfaction needs to be reported. Individuals or teams which solicit positive customer feedback need to be rewarded.

This process should be repeated until customer satisfaction is at an acceptable level, and then continuously repeated as customer needs may change over time.

11.1.3 Diffusing Challenging Customers

Despite the best efforts of the customer service staff to please all customers, there will be times where a particularly challenging customer will require assistance. A customer can be challenging for a number of reasons, the one addressed here is the upset customer. This customer has a problem, they don't have a solution, and are already upset coming into the telephone call. Part of providing effective customer service is understanding how to diffuse the challenging customer. In a short article, author Mor Assouline walks through four techniques for defusing an angry customer. His techniques are to [74]:

1. Listen
2. Apologize
3. Solve
4. Thank

Mich Solomon, in an article on Forbes, provides a separate but similar approach to turn an upset customer into a satisfied customer. Solomon's approach, titled AWARE, consists of the following techniques [75]:

1. Acknowledge
2. Widen
3. Agree
4. Resolve
5. Acknowledge

Both strategies share a common theme, put the customer first. Start by acknowledging the customer's concern. Regardless of how an individual feels, they should validate the customer's

concern. Listen to exactly what their concern is and try to recognize what they are upset about. Present yourself as a caring individual who is genuinely interested in their concern. The upset customer is typically going to have a lot to say, so allow them to speak. Just hear them out. At this point, feel free to either apologize, agree, or both. The goal here is to present empathy for the customer and their concern. After showing empathy, move on to actually solving the problem. This is ultimately what is going to diffuse the angry customer. Be sure to include the customer in your solution. Do not just leave them hanging and then present a solution which they are unhappy with. Rather, involve the customer. Walk through the solution with them as it is developed, regularly asking for their feedback. This will help ensure that the solution is satisfactory for the customer. At the end, be sure to thank the customer for their time.

It is very important that an individual avoids becoming upset or annoyed with the customer. This will only further escalate the situation and may lead to the loss of a customer. Maintain poise and composure despite the upset customer. The customer may be unruly, using foul language, etc. Do not view this as an opportunity to be upset with the customer or lecture them. Simply listen and identify the important details in the slurry of words which may be coming from the customer. While it is important to make the customer happy, do remember that no organization policies, including data classification policies, should be compromised in the process. Defusing the angry customer is one of the more challenging aspects of customer service, but a very necessary skill to run an effective customer service center.

11.2 Laboratory Exercise

In this laboratory exercise, the student will be introduced to the basic concepts of performing customer service as it applies to cyber defense competitions.

Performing Customer Service at Competitions

11.2.1 Specifications

This laboratory exercise will not require any technology. Rather, perform the exercises on your own, or with a partner(s).

11.2.2 Learning Objectives

- Creating a Voicemail Greeting
- Documenting Telephone Calls
- Responding to Caller Requests
- Maintaining Poise Under Pressure
- Responding to Technical Requests

11.2.3 Mapping to NIST Nice Framework

This laboratory exercise is intended to increase the student's skills at providing effective customer service. At cyber defense competitions, students can expect to have at least one indi-

vidual performing customer service tasks such as answering phone calls, responding to technical help desk requests, and documenting incoming requests. This laboratory exercise maps to the following KSAs from the NIST NICE Framework:

- Cybersecurity and Privacy Principles (K0004)
- Cyber Threats and Vulnerabilities (K0005)
- Electronic Devices (K0114)
- File Extensions (K0116)
- Industry Best Practices for Service Desk (K0237)
- Basic Operation of Computers (K0302)

11.2.4 Necessary Background and Expected Completion Time

This laboratory exercise does not have any necessary background. Because the exercises are open ended, expected completion time will be omitted. If being conducted in a course or other time-constrained environment, adjust the amount of time allocated for each challenge.

11.2.5 Challenges

1. *Make Your Help Desk Accessible*

   An organization needs to be accessible. One of the forms of contact that an organization has is the telephone. While it is ideal to have individuals on hand to field incoming phone calls at all times, this is not always possible. Rather than leave callers with a dead line at the end of an attempted call, having a pleasant voicemail greeting is in the interest of the organization.

   For this task, create a voicemail greeting for a fake organization. Compare your voicemail to the one included in the solutions to identify strengths and weaknesses of your voicemail greeting.

2. *Log Incoming and Outgoing Phone Calls*

   The importance of logs cannot be overstated, even when it relates to telephone calls.

Logging phone calls ensures that a record of all incoming and outgoing calls is kept. These records can be used to recall information from a previous call with an individual, document any requests made by the caller, and remind you to call an individual back when more information is available, among other things. Logs are only as effective as the details recorded; if your organization only logs the name of an individual and their telephone number, there is no meaningful information in the log which can be helpful later. Consider what details would be helpful in a telephone call logging sheet.

For this task, see the sample log sheet seen in table 11.1 and identify weaknesses. Then create your own sheet with information you feel would be helpful in a telephone call log. Compare the criteria you create with the criteria in the solutions.

Caller Name	**Time of Call**	**Receiving Employee**
John Doe	11:43	Trey

Table 11.1: *Sample Call Log with Entry*

3. *Practice Mock Phone Calls*

Phone calls can be intimidating in a professional environment, especially for an individual who does not have experience fielding telephone calls. This task will help individuals develop some familiarity with the process and increase comfort. There are three common types of phone calls which a team receives at competitions; these are the pleasant customer, the angry customer, and the nonsense customer. The pleasant customer will have a legitimate reason for calling and has patience for the individual answering the telephone. The angry caller has a legitimate reason for calling but lacks patience. This customer will have many demands and will require answers quickly. The nonsense caller does not have a legitimate reason for calling. They may be a spam caller or someone trying to waste the organization's resources. Despite this, one cannot be rude to the nonsense caller, however, efforts should be made to respectfully but quickly end the call.

For this task, there are three sample phone call scripts below. In the first scenario, your organization's billing system on the webpage is down. For the second scenario, the caller is upset that he cannot access the webpage. In the third scenario, the caller is trying to

sell you pizza rolls. Perform this exercise in groups of two, one individual acting as the caller, the other as the recipient. The scripts provide sample topics for the conversation, however, you may choose to make the conversation more natural by steering it in your desired direction. For tips on how to handle the angry caller, see the solutions.

> ***Incoming call from 555-111-2222****. Good morning, this is* ***Insert Name*** *from* ***Insert Organization Name****. I had a question regarding your website.* ***Response****. I noticed this morning that I wasn't able to access my electric bill. Is there something wrong with the website?* ***Response****. Oh I see, well, when will the issue be resolved?* ***Response****. Okay, thank you for the information. I will look forward to* ***Insert time/date*** *when I will be able to access my bill.* ***Response****. Thank you for your help, have a nice day!*

> ***Incoming call from 555-222-3333****. Hey, this is Jeff. What's wrong with your website bro?* ***Response****. Yeah look, I don't really care, I gotta pay my bill today or else I'll have a late fee.* ***Response****. Dude, are you telling me that you can't just fix it? What are you good for?* ***Response****. Can you just tell me when it'll be back up?* ***Response****. Hmm, yeah, I can't wait that long man! I need it sooner.* ***Response****. C'mon. Can't you just do some IT magic and make it work already? I pay for a service with you guys, I expect the page to be working!* ***Response****.* ***Continue upset questions with increased urgency as long as seen fit, the goal is actually to see if the recipient will crack under pressure****. All right, fine whatever. I'll be calling back later, with my boss on the line! It better be working by then!*

> ***Incoming call from 555-888-4444***. *Dude, this is Jenny from the pizza roll company, wanna buy some pizza rolls?* ***Response***. *But let me explain the deal to you first.* ***Response***. *Nah, I'm gonna go ahead and explain it anyway. So our current offer is 15 boxes of pizza rolls for only 25 dollars! The variety includes pepperoni, sausage, and combination.* ***Response***. *Now, you've been real nice, so let me tell you about a secret offer we have, right now we can get you 30 boxes of pizza rolls for only 50 dollars. So what do you say, are you interested?* ***Response***. *Okay, you drive a hard bargain, but how about 30 boxes for 45 dollars.* ***Response***. ***At this point, just keep them on the line as long as possible***.

4. *Answer Technical Help Desk Questions*

   Tech desk employees will have to be able to answer some degree of basic questions related to information technology. There are common tasks which they will have to be familiar with as well. Additionally, they will need to be familiar with organization policies well enough to be able to answer questions or at the very least defer to an appropriate party. For this task, answer the following sample technical questions:

   (a) A caller notices there are many new files on their machine which they don't remember being there previously. They are concerned that the files may be malicious. The caller is using a Windows machine with the standard file explorer. You want to know the extension of the files to make an initial determination if they are malicious.

   How you would direct the caller to tell you the file extensions?

   What are some common file types which may be malicious?

   (b) A caller notices that they are requested to create a new password. They are confused

why they need to do this. How would you explain to the caller why they need to change their password.

The same caller calls back later in the afternoon (logs would help you to know it is the same caller) confused why their new password isn't being accepted. Your organization has a new password policy with complexity requirements. Create the password policy and share with the customer what complexity requirements must be met.

(c) A caller is upset that their computer will have to restart in 15 minutes to apply updates. Explain to the caller why updates are important, and then suggest any steps the caller should take in the next 15 minutes to ensure no work is lost.

11.2.6 Solutions

1. *Make Your Help Desk Accessible*

   For this task, you will want to ensure that your voicemail greeting includes the following items.

   - The name of the individual or organization the caller has reached
   - An expression of apology for missing the call
   - Normal business operating hours
   - Alternative methods of contact (if applicable)
   - Request a message including the callers name (and organization if applicable), a call back number, and the reason for the call
   - A salutation

   Example of a High Quality Voicemail Greeting

> *Thank you for calling **XYZ Corp**. We apologize in advance for missing your call. Our business operating hours are **8 AM to 5 PM Pacific Time, Monday to Friday**. You may also contact us at **customerservice@xyzcorp.com**. Please leave your name, telephone number, and your reason for calling today, and we will return your call as soon as possible. Please record your message at the tone and have a wonderful day.*

2. *Log Incoming and Outgoing Phone Calls*

   A detailed log for incoming and outgoing phone calls should include the following items:

   - Incoming vs Outgoing
   - Name of caller (for outgoing, this is the employee making the call)
   - Name of callee (for outgoing, this is the person being called)
   - Date of call
   - Time of call (either use 24 hour time or include AM vs PM)
   - Number of caller (if incoming)
   - Number of callee (if outgoing)
   - Description of call
   - Notes

3. *Practice Mock Phone Calls*

   Target criteria for each of the mock calls.

   Pleasant Caller

   For this caller, be sure to thank them for their time. Explain what the issue is and when you expect it to be resolved. Express your apologies for their having technical issues. If possible, make note to give them a call back when the service is available again.

   Angry Caller

For this caller, it is crucial that you stay calm and poised. They are going to try to get you to get angry in retaliation. This caller will have a lot to say, and isn't interested in hearing much. Rather than try to reason with this caller, simply reassure them that you understand their issue and apologize for the inconvenience. This caller may get personal, but just maintain composure. Answer objectively and appropriately. It can be easy to get pressured by this sort of caller to either provide either false information (I think the service might be back up later this afternoon) simply to provide an answer to their question, or to provide confidential or privileged information (the service is down because we were hacked). Remember that an upset phone call is only a temporary event and will be over soon, and focus on making the customer as happy as possible, even if they appear unhappy.

Nonsense Caller

For this caller, be sure to be respectful and not dismiss their call. Yes it may be annoying, and you may have other more time sensitive matters to attend to. Simply hear them out, answer any questions appropriately. Make an effort to end the phone call, but do not just hang up on the individual. If the call is getting too long, request that the customer call you back later. If you absolutely must leave the call, express your sincere apology for having to end the call abruptly.

4. *Answer Technical Help Desk Questions*

   (a) Direct the caller to open file explorer and navigate to the folder containing the new files. Explain to them how to view the files in **Details** view. If file type is not a visible option, instruct them to enable the file type check box in the column names. Also have them note the extension on the end of the file. A non-exhaustive list of common file types which may be malicious includes executable file types (.exe, .bat, .msi, etc.), Microsoft Office file types (.docx, .pptx, .xlsx), and script file types (.py, .sh, .pl, etc.).

   (b) Explain to the caller that because passwords protect our accounts from being accessed by an unauthorized user, it is important to regularly change passwords. In

the event that a password has been compromised, a password change can prevent the unauthorized user from maintaining access to our assets. Explain to the caller that password policies are in place to prevent users from creating simple passwords which can easily be cracked by a malicious user. Explain your created password policy and why you chose what you did (i.e. length vs. complexity).

(c) Explain that new vulnerabilities are discovered regularly and updates are regularly released to mitigate against the aforementioned vulnerabilities. Without these updates, the user's devices remain vulnerable and be the target of a cyber attack. In order to ensure no work is lost, suggest that the caller save any documents he/she is working on at the moment and close any applications which may be running.

Chapter 12: Organizational Management Tasks Applied at Competitions

12.1 Background of Organization Management Tasks

This section discusses, in detail, concepts related to organizational management which are commonly needed at cyber defense competitions.

12.1.1 Creating a Comprehensive Information Classification Program

Organizations today deal with so much data. Often, it can be difficult to determine what data is important, what data is not important, and what data is considered sensitive. To ensure that sensitive data is not released to unauthorized individuals, creating a comprehensive information classification program can be helpful. Andrew McCreath notes the following in a Computer Weekly article [80]:

> *"Data classification best practices enable organisations to store their data in line with compliance controls, thereby reducing any risk to the business in the event of an audit or legal discovery."*

The legal issue is just one of many reasons to implement a data classification program. Additionally, data classification can be useful in ensuring that the concept of least privilege is achieved, that is, that only individuals with a need to know have access to designated information. A comedic, yet relevant example of this exists in the children's television show, "SpongeBob SquarePants" [81]. In the show, the owner of a restaurant has a secret formula for creating their signature item, the *Krabby Patty*. Throughout the series, various individuals attempt to gain access to the formula, though the owner ensures that information is protected. Even his own employees don't know the secret formula because they do not need to in order to perform their function. Similarly, organizations should ensure that privileged information is limited to only those individuals who require access.

This further begs the following two questions:

- What levels of information classification should an organization have?

- How does an organization determine what access level an individual should have to information?

The answer to the first question is very subjective. Every organization will handle information differently, and different types of information. Depending on the type of information, different classification levels will be needed. For example, in healthcare organizations, protected health information (PHI) are present whereas at a automobile production company, engine design details exist. This leads one to need to go through a full process to determine how to set up an appropriate information classification program. Generically, there are three high level categories of classification according to an article in Strong DM relating to SOC 2 Compliance [79]. Author Brian Johnson presents these three high level categories in descending order of sensitivity. These levels are Confidential, Internal, and Public. Public data is information with no protection. This data is available openly to the public and may include items such as what products the company is producing. Internal data is protected within the organization. These data should not be discussed publicly but may freely be discussed internally. These data may include employee salaries. Lastly, confidential data is confined to select individuals with a strict need to know. These information may include employee and client information including social security numbers.

In a Sirius Edge article, authors Thomas Eck and Anne Grahn outline seven steps to creating an effective information classification program [78]. These steps are as follows:

1. **Complete a Risk Assessment**: Determine what regulatory and contractual requirements the organization has related to data confidentiality.

2. **Develop a Formal Classification Policy**: Determine the categories of data classification. Limit to three to five categories to avoid unnecessary confusion over high level of granularity.

3. **Categorize Data Types**: Determine what protected data the organization manages. This means identifying what data the organization maintains, creates, or handles. Knowing the data and assets allows the organization to determine classification levels.

4. **Discover Location of Data**: Now that the types of data have been identified, the organization needs to identify where the data are located. This may be physical location of data (storage room in the room 104 of the east wing) or digital (credit cards folder on the private share).

5. **Identify and Classify Data**: Formally classify the data. Make it known to the organization that a policy is in place and must be followed.

6. **Enable Security Controls**: Implement the program. This may require limiting access to data digitally using group policy type software or physically by granting access to specific areas. Additionally, the organization can implement encryption for digital data as well as air gaps for physical data.

7. **Monitor and Maintain**: Having implemented the program, it is critical that the program is maintained. Regularly monitor both digital and physical accesses to high sensitivity documents and ensure that the access was from an authorized individual. Be sure to regularly reevaluate the classification level of data as well as the level of access an individual within the organization has.

Following the seven step process, an organization should be able to implement a fairly comprehensive information classification program.

12.1.2 Selecting the Appropriate Organizational Structure

An effective organization requires an appropriate organizational structure. Organizational structures refer to the configuration of relationships between positions and teams within an organization. According to articles from Smart Draw and Point Park University, four common organizational structures include [90, 89]:

- Functional Top-Down
- Divisional
- Matrix

- Flat

The functional top-down structure is hierarchical and very much based on chain of authority. In this structure executives rest at the top of the structures, with senior management below, followed by junior management, team leads, etc. This organizational structure excels in it's ability to group similarly skilled individuals in teams which allow a focused approach to problems. However, this structure leads to a lack of communication between groups and poor transparency up the structure.

The divisional structure has the organization split into various divisions which operate as sub organizations in some sense. In this structure, the individual divisions have a high level of independence and may not require higher level authority to make decisions. However, in this structure, individuals working in similar capacities in different divisions may have little collaboration.

The matrix structure has its teams split based on projects or products. Individuals may work as part of a project with a project lead, but will typically also have a function manager they report to. This structure excels in its ability to facilitate communication and provide a dynamically shifting work environment. This structure struggles in coordination at times. Individuals may belong to multiple projects and thereby report to separate project leads, between whom their may be little communication.

The flat structure has very little to no hierarchy. All individuals report equally to a single authority. This structure enables a very individually managed team where individuals work independent of others and provide autonomy for each employee. This structure does struggle in that by providing each employee autonomy, there can be disagreements on how to proceed with a task without intervention of the single higher authority.

Which organizational structure should be implemented is entirely dependent upon the needs of the organization, what type of products/services it offers, and the size of the organization, among other things.

12.1.3 Choosing an Effective Team

An organization is only as successful as its staff. Ensuring that an effective team of staff members has been chosen can be the difference between a successful organization and failed organization. Articles from BrightWork [92] and AboutLeaders [91] outline tips for selecting an effective team. They suggest hiring individuals who are able to communicate effectively. Individuals who can listen to others in a meaningful manner but also respond eloquently are invaluable. These individuals can help facilitate cooperation internally but also interact with sponsors and prospective clients.

Additionally, skilled project managers should be hired. The organization will need an individual(s) who possess strong leadership skills and can manage small to large teams to lighten the burden on any single individual within the organization.

Another suggestion is to remain strictly objective in the hiring process. It can be tempting to hire an individual who is family friends with a project lead, but if a better candidate applies for the role, ensure that the best candidate is hired.

It is important to be mindful of the budget your organization has as well as what needs it has. Select the team accordingly to ensure the most effective team.

12.2 Laboratory Exercise

In this laboratory exercise, the student will be introduced to organizational structures, team selection, and simple data classification policy.

Organizational Management Tasks Applied at Competitions

12.2.1 Specifications

This laboratory exercise will not require any technology. Rather, perform the exercises on your own, or with a partner(s).

12.2.2 Learning Objectives

- Understanding Organizational Structures
- Levels of Data Classification
- Selecting an Effective Project Team

12.2.3 Mapping to NIST Nice Framework

This laboratory exercise is intended to increase the student's skills in business management. At cyber defense competitions, organizing a team and operating an organization is scored in addition to the technical tasks. This includes organizing the team, creating a data classification program, and selecting an effective team. This laboratory exercise maps to the following KSAs from the NIST NICE Framework:

- Cybersecurity and Privacy Principles (K0004)
- Cyber Threats and Vulnerabilities (K0005)
- Organization's Information Classification Program (K0287)
- Apply Cybersecurity and Privacy Principles to Organizational Requirements (S0367)
- Apply Cybersecurity and Privacy Principles to Organizational Requirements (A0123)

12.2.4 Necessary Background and Expected Completion Time

This laboratory exercise does not have any necessary background. Reading documents on organizational structure types may be helpful as well as understanding basic data classification concepts. Because the exercises are open ended, expected completion time will be omitted. If being conducted in a course or other time-constrained environment, adjust the amount of time allocated for each challenge.

12.2.5 Challenges

1. *Using an Effective Organizational Structure*

   Organizational structures can help determine how individuals within an organization interact with each other, as well as how teams are managed. Four common organizational structures are the **functional top-down, divisional, matrix,** and **flat** structures.

   For this task, do some research about the four organizational structures above, read the descriptions of two mock organizations seen in table 12.1, and determine which organizational structure fits best for the two.

2. *Creating an Comprehensive Data Classification Program*

   Information classification is necessary for organizations which handle sensitive data. Your organization currently has no information classification program, but has various levels of sensitive documents.

   For this task, create an information classification program for your organization. Explain

MeCorp Energy	At MeCorp, we believe giving our employees autonomy. The classic hierarchical structure, that's not what we stand for. Our organization has sponsors for various projects, and project managers. Our project managers choose various other employees to work on their projects. Employees may be working on multiple projects at any given time. Project managers individually organize their teams how they see fit, assigning team leads to provide granularity of the organization.
XYZ Technologies	At XYZ Technologies we stand by the traditional managerial methods. Our executives sit atop the organization. Senior and junior management follow. Below management we have our project leads then manage individual teams. This method allows us to follow a direct chain of command with decisions bubbling up to an appropriate level.

Table 12.1: *Sample Organization Mission Statements*

the levels of classification as well as how the different levels of information should be managed.

3. *Choosing the Right Team*

   An organization needs to ensure that the right team is selected to meet its needs. A strong team needs a variety of individuals who possess strengths in different areas, technical and non-technical, managerial and non-managerial. In reality, it is very rare that an organization will hire a full team at one time; however, for the purposes of this exercise, you will be performing the task of selecting a full team.

   For this task, read the required personnel traits seen in table 12.2. Your mock organization will have a budget of $400,000 to hire the required personnel. You will also be given a list of candidates, each candidate having a certain level of strength in multiple categories, and the cost of hiring said employee. Use these information to select the most qualified team to fit the organization's needs while staying under budget.

   The traits that each individual possesses are Leadership, Independence, Teamwork, Technical, Soft Skills.

   Requirements:

   This team needs needs a strong project manager, this individual needs to have high leadership and teamwork traits. The team also needs at least two programmers. Our pro-

grammers need strong technical ability as well as the ability to work well as a team for some projects but also develop independently for others. Our team would like to be able to reach out to potential clients with an individual who bridges the gap between technical and non-technical. This individual will need to have very strong soft skills but be well versed technically as well. Teamwork is important to this team, we want to have a very strong team culture.

Name	**Cost**	**Leadership**	**Independence**	**Teamwork**	**Technical**	**Soft Skill**
Joseph	$170,000	6	7	6	10	3
Marianne	$70,000	7	6	5	8	3
Tessa	$75,000	8	4	6	5	5
Chris	$65,000	5	5	6	7	5
Gordon	$80,000	7	4	5	6	7
Trey	$150,000	10	5	7	6	6
Jordan	$40,000	3	4	3	6	5
Shawn	$30,000	2	4	6	3	9
Harold	$45,000	4	4	4	7	7
Brittney	$200,000	10	8	9	8	8
Jayci	$70,000	3	5	6	9	4
Brooke	$60,000	6	4	7	6	8

Table 12.2: *Prospective Employee Traits*

12.2.6 Solutions

1. *Using an Effective Organizational Structure*

   For MeCorp Energy, the flexibility and autonomy of the organization suggest that a Matrix organizational structure would be a good fit.

   For XYZ Technologies, the traditional hierarchical design and employing a strong chain of command suggest that a Functional Top-Down organizational structure would be a good fit.

2. *Creating an Comprehensive Data Classification Program*

   You will want to ensure that your data classification program does not contain to many

layers of granularity. Maintaining a classification program with a single layer is convenient but insecure; conversely, one with too many layers is secure, but very inconvenient. Your structure should include 3-4 layers. Common programs use layers such as Public, Internal, and Confidential or Public, Internal, Confidential, and Secret. You will also want to ensure that appropriate information is being kept in each layer. This will vary by organization. For example, a restaurant may value its recipes more than its employee records (not suggested for legal reasons) while a medical company may value its patient records at highest priority (for HIPAA purposes).

3. *Choosing the Right Team*

   This exercise can be performed many ways, and there may be multiple high quality teams which can be selected from the team above. The main takeaways should be that the company requires four employees; 1 project manager, 2 programmers, and 1 sponsor representative.

   Looking at the candidate pool, we can break the candidates into which categories they possess an affinity for trait wise.

   Project Manager Candidates: Trey and Brittney both have high leadership traits and both have above average teamwork traits.

   Programmer Candidates: Joseph, Marianne, Brittney, and Jayci all have high technical traits, as well as average to above average independence and teamwork traits.

   Sponsor Representative Candidates: Shawn, Brittney, and Brooke all have high soft skill traits. Brittney and Brooke have above average technical trait while Shawn has a below average technical trait.

   This sort of problem has many known variants in computer science, one of which is known as the Knapsack Problem. In essence, the goal is to select the highest value items (in this case candidates), while remaining under a threshold (in this case the budget). While there may be other team selections, potentially better, one example, along with a justification follows.

   For the following team, Brittney has been selected as the project manager, Joseph and Marianne have been selected as the programmers, and Brooke has been selected as the

sponsor representative.

The choice for project manager was made by virtue of Brittney having more strength in the teamwork trait. Although Brittney does have a higher cost, the increased teamwork was valued at a higher priority.

The choice for Marianne and Jayci was made based on cost. Although Joseph was a stronger candidate in terms of trait values, the cost was not worth the trade off. Brittney was no longer considered on account of already being selected for the project manager position.

The choice for the sponsor representative was made because Brooke had a much higher technical trait, which was necessary for this position. Brittney was again not considered on account of having already been selected for project manager.

As stated earlier, this is not the only solution to this problem, and may not even be the most optimal solution. Consider writing a computer program which calculates the most optimal team given the constrains of this problem.

Chapter 13: Introduction to Incident Management and Response

13.1 Background on Incident Management and Response

This section discusses, in detail, concepts related to incident management and incident response as they apply at cyber defense competitions.

13.1.1 Types of Potential Incidents

When dealing with cyber threats, it can be challenging to know where to begin. A good place to start is to identify what type of threat is present. The Commonwealth of Massachusetts and the U.S. DHS's CISA provide descriptions of various different types of cyber threats [84, 85]. As noted by the Commonwealth of Massachusetts, common threat vectors include [84]:

- Malware
- Phishing
- Denial of Service
- Ransomware

In addition to the type of threat, the source of the threat should be noted. The U.S. DHS's CISA outlines multiple potential sources for cyber threats including [85]:

- Foreign Intelligence Services
- Independent Hackers
- Insiders
- Terrorists

Any of the types of attacks can be carried out by any of the potential sources.

Malware is described as malicious code or software. The individuals responsible for creating malware typically aim to compromise the overall security of the target system. Malware can range from very obviously present on a machine to extremely covert. Complex malware have the ability to steal data, compromise systems, and even take down power grids.

Phishing is a relatively new form of attack which takes advantage of email as an attack vector. Phishing attacks can be generic or targeted. An example of a generic phishing scheme is the Nigerian Prince. In this scheme, the unsuspecting user receives an email from a prince claiming to have millions in assets which they will send you if you wire them a small sum of money to be released from prison (or something similar). These schemes, while unsavory, do not necessarily have the ability to cause significant damage. Targeted phishing attacks on the other hand take advantage of information about the target which help convince the target that the sender is legitimate. These emails may appear to come from a trusted sender and typically request that you either click on a link or download a file. The unsuspecting target is unaware that the link may route them to a website which will attempt to solicit confidential information under the guise of a legitimate source such as the target's bank, or download a file which executes malicious code upon being opened.

Denial of Service (DoS) attacks perform exactly what they sound like; they attack a target such that the target is no longer accessible. In the case of a web page, DoS attacks may spam the page with so much traffic that valid users can no longer access the page. DoS attacks are a significant inconvenience to the target and require attention, which then allow malicious attackers to commit other acts such as intrusions or fraud while the response teams are distracted by the DoS attack.

Ransomware is another relatively new type of attack. Typically, the malicious actor uses malware to encrypt the targets assets or file system. The malicious actor then requires a sum of money to decrypt the assets.

13.1.2 Dealing with Insider Threats

Insider threats are those which occur internal to an organization. Three common archetypes for insider threats, according to a Sirius Edge article, are Mistake-Makers, Malicious Insiders, and Imposters [77]. Mistake makers are described as individuals within the organization who fall for phishing schemes or otherwise become an a means of access to data. This individual is not necessarily involved in the malicious act, but rather, their carelessness makes them a vul-

nerability. Imposters are described as malicious actors who use legitimate credentials (typically stolen) for employees of an organization to access data legitimately. The malicious insider is a current or previous employee or other related figure to the organization, such as a contractor, who exploit the fact that they have authorized access to data [77].

Insider threats are challenging to defend against, especially in the case of the malicious insider, because the behavior is normal. An authorized individual is accessing data which they access every day. Although the technical behavior is not anomalous, there are certain measures which an organization can take to help combat against insider threats.

In an article from Sage Data Security, author Becky Metivier provides four tips for detecting insider threats [76].

1. **Be Aware**: An organization needs to know where their most sensitive data is located and monitor access to that data. This means that the organization may need to perform an assessment of the data, determine what is valuable, and keep record of the location of that data. Which individuals access the data, when they access the data, and how they access the data should all be recorded.

2. **Change Things Up**: An organization needs to be modular. Criteria such as the location of data, authorized individuals and stewards of the data, and monitoring of the data should be regularly rotated. Keeping the system under constant change means that a single individual or team never has extended access to the data, which can make it easy for them to regularly exfiltrate small amounts of data so as to go under the radar.

3. **Know Indicators of Compromise**: There are many indicators of compromise which can lead to the discovery of an insider threat. One of the way that insider threats are carried out is to exfiltrate data. This can be detected by monitoring data transfer. An insider may transfer anomalous amount of data, whether it downloaded onto an external drive, sent across email, or uploaded to a file sharing service or cloud service. Monitor access logs, checking if individuals have been accessing assets anomalously (outside of normal hours, special access areas without authorization), if terminated employees are accessing organization systems, or if an individual who has been transferred internally is accessing

previously authorized assets.

4. **Implement Security Technologies**: Knowing the indicators of compromise is only valuable if there are measures in place to protect against the threat. Regularly updating employee accesses will help to prevent against a terminated or transferred employee from having residual access to previous assets. For critical assets, ensure the data is encrypted, this will keep it from being useful if exfiltrated.

If suspicion rises that an organization may have an insider threat, formal investigative work will need to be done. Accusing an employee of being a potential insider can not come lightly. Documentation, logs, evidence are key; if approaching a potential insider, leading with evidence will help support any claims made [76].

13.1.3 Responding to Cyber Incidents

Incidence response is not a new topic, nor is it poorly documented; various organizations have explicit response plan steps. An article from AlienVault lists the steps that NIST and SANS use for incident response [62]. Both have very similar models, including the following steps:

1. Prepare for an Incident
2. Identify and Analyze an Incident
3. Contain the Incident
4. Eradicate the Incident
5. Recover from the Incident
6. Perform an Analysis Post Incident

Exabeam lists best practices for incident response plans [72]. The best practices include using automation, leveraging playbooks, and testing the incident response plan [72]. In a time where there is so much data, it can be difficult to perform effective data analysis to detect

anomalous activity which may be an incident. Using automation can help cut out some of the noise, making it easier to identify the anomalous events.

Hacking groups, nation state actors, and other malicious actors tend to follow similar patterns on low profile attacks. Playbooks can be used to prepare a response to incidents which perform a series of steps on a prospective malicious incident.

As with any plan, an incident response plan needs to be tested to determine its efficacy. Carry out a mock incident and determine how the incident response plan fared against the mock incident.

A proper incident response plan will help ensure that an organization is prepared for incidents and can swiftly minimize the impact. Other sources for incident response plans include the U.S. Department of Homeland Security's National Cyber Incident Response Plan [59] and NIST's Computer Security Incident Handling Guide (SP 800-61 rev.2) [60].

13.1.4 Documenting Cyber Incidents

As with anything else in life, documentation is critical. Having detailed and accurate cyber incident documentation will help the response team to perform their role swiftly and effectively. There are many quality resources which walk through cyber incident reporting and documentation.

One suggestion, from a Turn Key Technologies article, is to build up a tiered reporting process [70]. Author Tony Pugielli explains that having separate reporting procedures for different groups within an organization can lead to confusion regarding which procedure to follow and may slow the documentation process [70]. By using a comprehensive documenting procedure which applies to all groups within the organization, response teams can quickly document any necessary information, following the universal documentation process.

Having a cyber incident documentation template can help ensure that all appropriate information is recorded. Any necessary evidence from the incident can also be collected. In the event that the incident requires legal dispute, having proper documentation is crucial, according to a Digital Guardian article [71]. A template will help ensure that critical information, such as

the time the incident was detected or the systems impacted, are noted. The template could be based on an organization wide policy for incident documentation which ensures that sufficient information is record to be presented in a legal court [71].

In a Computer Weekly article, author Dinesh Bareja provides a list of best practices for security incident management [73]. Bareja suggests having a policy and procedure in place to ensure that as information becomes available, it is documented in a specific way, including specific details. It is also suggested there are clearly defined roles for individuals in the incident documenting procedure. Bareja notes that carrying out regular training with sample incidents can help ensure that an organization's employees stay sharp on the skills and allow management to assess if any employees need clarity on their role in the reporting process [73].

Having clear and defined information which should be collected and a template will ensure that employees do not leave out necessary information or include unnecessary information. An organization can develop its own incident report template or adopt pre existing templates. Two existing templates for incident reporting are the U.S. Department of Homeland Security's FEMA form ICS 201 [58] and the U.S. Department of Homeland Security's US-CERT Incident Reporting System [61].

13.1.5 Briefing Appropriate Audiences on Cyber Incidents

Organization suffer from cyber incidents, it is inevitable. However, how an organization documents the incident, responds to the incident, and briefs the appropriate audiences will determine the long term impact of the incident on the organization. Audiences can range from a small group within an organization to the public; regardless of the size of audience, an appropriate briefing of the event is necessary. Consider the following suggestions when preparing a briefing.

In a document from the Massachusetts Institute of Technology titled *Guidelines for Effective Briefings*, it is suggested that the presenter determine the medium for the briefing [82]. Consider the audience for whom the briefing is being prepared. Common mediums for briefings include spoken, email, and pre-recorded video. Another tip is to follow an organized script of topics.

A suggested order for topics is to begin with background and context of the incident, follow up with an appropriate amount of detail regarding the organizations response, and finish with a conclusion of the current state of the incident and what the organization has done to prevent a similar incident in the future [82].

In a Chron article, author Jackie Lohrey stresses the importance of an effective Q&A period. After an incident, the audience will surely have questions. Lohrey notes that the distinguishing factor of a successful briefing is an effective question and answer session. There may be some common questions which can be anticipated, such as the source of the incident, affected systems, what it means for clients, etc. The organization should prepare well thought out answers to these common questions. For the more challenging or random questions, an organization should ensure that the individual giving the briefing is able to articulate well, possesses strong stage presence, and speaks with assurance [83]; even if the question cannot be answered directly, these qualities will speak to the audience.

13.2 Laboratory Exercise

In this laboratory exercise, the student will be introduced to the stages of incident response and apply incident response concepts to mock incidents.

Introduction to Incident Management and Response

13.2.1 Specifications

This laboratory exercise will not require any technology. Rather, perform the exercises on your own, or with a partner(s).

13.2.2 Learning Objectives

- Identifying Types of Incidents
- Familiarity with Stages of Incident Response
- Assessing Incident Response Plans
- Applying Incident Response Plans
- Documenting Cyber Incidents
- Qualities of an Effective Briefing

13.2.3 Mapping to NIST Nice Framework

This laboratory exercise is intended to increase the student's skills in the area of incident response. At cyber defense competitions, students can expect to suffer from a range of cyber incidents. Skills including documenting, responding to, and briefing on cyber incidents are necessary to perform well at competitions. This laboratory exercise maps to the following KSAs from the NIST NICE Framework:

- Cybersecurity and Privacy Principles (K0004)
- Cyber Threats and Vulnerabilities (K0005)
- Hacking Methodologies (K0310)
- Documenting Reported Incidents, Problems, and Events (K0317)
- Recognize Types of Vulnerabilities (S0078)
- Accurately Define Incidents, Problems, and Events (A0025)

13.2.4 Necessary Background

This laboratory exercise does not have any necessary background. Familiarity with cyber incidents and the incident response process may be helpful but is not required. Because the exercises are open ended, expected completion time will be omitted. If being conducted in a course or other time-constrained environment, adjust the amount of time allocated for each challenge.

13.2.5 Challenges

1. *Types of Incidents* There are many different types of cyber incidents. Each with specific capabilities and purposes.

   For this task, you will be matching types of cyber incidents, seen in table 13.1, with their descriptions, seen in table 13.2.

2. *The Stages of Incident Response*

   The incident response process is well documented with many suggested procedures to go

Malware
Phishing
Denial of Service
Ransomware

Table 13.1: *Types of Incidents*

This type of incident typically uses email as its medium of attack. The malicious user sends an email pretending to be a legitimate sender and aims for the target to either click a malicious link or download a malicious file.
This type of incident typically involves locking the targets assets by encrypting their hard drive. The malicious user promises to unlock the targets assets upon receiving a sum of money.
This type of incident typically involves flooding a targets asset with so much traffic that it is unable to operate properly. The malicious user sends a mass of network traffic to the target system until it crashes or is otherwise unable to perform its intended task.
This type of incident is categorized by malicious code or software. The malicious user crafts special code to compromise the security of the target system in hope to steal data or damage the target system.

Table 13.2: *Incident Type Descriptions*

from preparing for the incident to post incident analysis.

For this task, read the mock incident response seen in table 13.3. It will have the incident response steps in it using specific terminology. Using the scenarios, try to identify the stages of incident response. The stages in the plan being referenced map to the NIST and SANS incident response frameworks. Compare your list with the list in the solutions.

3. *Assess an Existing Incident Response Plan*

   Organizations do not always have the best incident response plan in place. Sometimes this can lead to certain actions not being performed and specific evidence not being collected. In addition to letting certain aspects of the incident go unattended to, this can lead to the organization not having enough evidence or information to take the incident to court if the suspected perpetrators are determined.

   For this task, read the stages of the poorly designed incident response plan with vague terminology below. Using these stages, explain the steps you would take to respond to the incidents described below. Then assess the strengths and weaknesses of the poorly

Response	At MeCorp Energy, we recently had a cyber incident. Our team had prepared for a potential incident by contracting with a cyber assessment team who performed an assessment of our infrastructure to determine where our weak points were. Our organization identified that the attack was a standard Denial of Service attack. Further analysis revealed that the attackers used a SYN flood attack which affected our NA-West web servers. In an effort to contain the incident, upon detection, we took our NA-West server offline to prevent the incident from spreading. Once we were able to figure out how the attackers were accomplishing the SYN flood, we were able to use additional firewall rules to eradicate the attackers from our systems. We have since began working to improve our firewall rules to recover from the incident. Our team is currently performing a post incident analysis to identify the attackers and implement any findings into our incident response plan.

Table 13.3: *Sample Incident Response*

designed incident response plan.

Poorly Designed Incident Response Plan

(a) Detect the Incident

(b) Get Rid of the Threat

(c) Involve Law Enforcement

Mock Incident #1

XYZ Technologies is reporting on a recent cyber incident. On the 19th of September, 2019, at 7:43PM, our operations center detected an incident. A staff member of our administrative team received an email from their internet service provider on a personal laptop. The email was part of a phishing scheme which included a downloadable billing statement. Upon downloading the billing statement, a malicious file began running on the personal laptop. Unfortunately, the laptop was connected to the XYZ Technologies internal network, causing the malware to spread to other machines on the network.

Mock Incident #2

Delicate Deserts and Drinks has received confirmation that a recent global ransomware campaign has affected our systems. At approximately 4:27AM on April 13th, 2018, one

of our employees documented an inability to open our inventory spreadsheet for the day. Upon inspection, the file system on the affected machine is encrypted and requires a payment of $500 equivalent in Bitcoin. At present, no other machines appear to be infected, though customers commonly bring personal laptops and connect to the same network our systems are connected to. Our doors open at 7:00AM.

Mock Incident #3

The public relations manager at Ponderosa Regional Medical Center reported a loss in power in certain parts of the hospital at 5:34PM on February 11th, 2003. At present, the exact cause is unknown, though there is suspicion that the outage is the result of a cyber attack. An intern at the hospital reports that they found a thumb drive outside the hospital earlier that morning and plugged it into their workstation on the cardiac floor. Shortly thereafter, the cardiac floor lost power. Backup generators have been activated to avoid loss of life.

From here, using the poorly designed incident response plan, discuss how you would respond to the incident.

4. *Develop a New Incident Response Plan and Document*

   Incident response is not static. It is constantly improving, with new takeaways being implemented after each incident encountered.

   For this task, given the same mock incident from task 3, design a new incident response plan. Use this new plan and explain what steps your organization would take to respond to the same incidents in task 3. Explain why your plan is improved from the poorly designed plan. Additionally, document any details from the incidents that you feel are necessary.

5. *Brief Audiences on Incidents*

   After an incident, briefing an appropriate audience may be necessary. Briefings need to include enough detail to satisfy the audience but not so much that they may interfere with any ongoing incident response processes.

   For this task, practice briefing an audience based on the outcome of your discussion for

carrying out the new incident response plan. See the solutions for some key points to note.

13.2.6 Solutions

1. *Types of Incidents* The correct matching of types of incident and description are shown in table 13.4.

Phishing	This type of incident typically uses email as its medium of attack. The malicious user sends an email pretending to be a legitimate sender and aims for the target to either click a malicious link or download a malicious file.
Ransomware	This type of incident typically involves locking the targets assets by encrypting their hard drive. The malicious user promises to unlock the targets assets upon receiving a sum of money.
Denial of Service	This type of incident typically involves flooding a targets asset with so much traffic that it is unable to operate properly. The malicious user sends a mass of network traffic to the target system until it crashes or is otherwise unable to perform its intended task.
Malware	This type of incident is categorized by malicious code or software. The malicious user crafts special code to compromise the security of the target system in hope to steal data or damage the target system.

Table 13.4: *Matching Incident Types to Descriptions*

2. *The Stages of Incident Response*

   A combination of the NIST and SANS incident response plans results in the following steps for incident response:

   (a) Prepare for an Incident

   (b) Identify and Analyze an Incident

   (c) Contain the Incident

   (d) Eradicate the Incident

   (e) Recover from the Incident

   (f) Perform an Analysis Post Incident

3. *Assess an Existing Incident Response Plan*

   This plans strengths are that it vaguely suggests that the organization should identify and analyze the incident (detect the incident) and eradicate the incident (get rid of the threat). The terminology needs to be more concrete here. The step to involve law enforcement should not be on the list. Law enforcement should only be involved as necessary.

4. *Develop a New Incident Response Plan*

   Be sure to note the following items are noted and documented for each incident:

   - Date
   - Time
   - Detecting Individual
   - Systems Impacted
   - Description of Incident
   - Steps Taken in Response

   Incident #1

   Be sure to question whether or not personal devices should be connected to the same network as business assets. Your discussion should include a plan to hold phishing awareness training after the incident.

   Incident #2

   Be sure to question why customers are not connecting to a guest network instead of the business network. Your discussion should include conversation regarding whether or not the ransom should be paid. Also comment on what steps should be taken with doors opening in less than two hours from the detection of the incident.

   Incident #3

   Your discussion should include a plan to hold training on removable device hygiene. Also discuss what actions may need to be taken since loss of life is a potential consequence in this case.

5. *Brief Audiences on Incidents*

   Criteria which should be noted in the briefing include:

   - Date
   - Broadly, what systems are impacted
   - Broadly, what the impact was
   - In detail, what has been done so far in response
   - In detail, the plan going forward

   The briefing should begin with a brief description of the incident, then cover the criteria above, end the session with a question-answer period. Be sure not to give away any details which may be considered confidential according to your organizations classification program or any information which is too technical, the audience is likely more interested in what the incident means for them, what has been done to fix it, and how you can assure them it will not happen again.

Chapter 14: Conclusions

Cyber defense competitions have shown to be effective at preparing participants for integration into the workforce. CYOTEE fills the need for targeted preparation material with respect to cyber defense competitions. By completing the laboratory exercises in CYOTEE, participants will develop knowledge, skills, and abilities which directly map to those indicated in the NIST NICE Cybersecurity Workforce Framework. The laboratory exercises provide participants with the opportunity to perform various technical tasks which are motivated by tasks seen at competitions. Skills such as securely using MySQL, appropriately configuring Active Directory, and creating secure web applications are necessary to effectively run an enterprise IT environment, especially at cyber defense competitions. The discussion-based laboratory exercises provide the participants with the opportunity to discuss topics which are less commonly thought of as technical, but are equally important in an enterprise IT environment. Topics such as customer service, organizational management, and incident management are critical to the successful operation of an organization and are included at cyber defense competitions. CYOTEE exercises are available for free in the project GitHub repository: **https://github.com/CenterForSecureAndDependableSystems/CYOTEE**. The exercises can also be modified freely as needed.

References

[1] "CCDC Regionals", *nationalccdc.org*, 2018. [Online].

[2] United State. Cong. *Recognizing the National Collegiate Cyber Defense Competition for its now five-year effort to promote cybersecurity curriculum in institutions of higher learning.* H.Res.1244. 111^{th} Cong. (2009-2010). [Online].

[3] "About - CyberForce Competition™", *Cyberforcecompetition.com*, 2018. [Online].

[4] "CyberForce Competition ™- A DOE Cyber Defense Competition", *Cyberforcecompetition.com*, 2018. [Online].

[5] Executive Order no. 13800. 11-May-2017.

[6] T. Nuth, "Cybersecurity concerns intensify for critical infrastructure worldwide", *www.SecurityInfoWatch.com*, 2018. [Online].

[7] A. Pattanayak and M. Kirkland, "Current Cyber Security Challenges in ICS", *2018 IEEE International Conference on Industrial Internet (ICII)*, 2018. Available: 10.1109/icii.2018.00013.

[8] "AFA CyberPatriot Website", *UScyberpatriot.org*, 2018. [Online].

[9] "SEED Project", *cis.syr.edu*, 2018. [Online].

[10] L. Cridlin, "The Importance of Hands-On Learning", Laser Institute of America, 2007.

[11] K. Prichard and R. Sawyer, *Handbook of College Teaching.* Westport (Conn.): Greenwood, 1994.

[12] S. Hidi, "Interest, Reading, and Learning: Theoretical and Practical Considerations", *Educational Psychology Review*, vol. 13, no. 3, pp. 191-209, 2001.

[13] B. Clark, *RTFM: Red Team Field Manual*, 1^{st} ed. CreateSpace Independent Publishing, 2014.

[14] National Collegiate Cyber Defense Competition, *NCCDC Regions.* 2018.

[15] Pink Elephant Unicorn. *Pink Elephant Unicorn. Pacific Northwest National Laboratory.* [Online].

[16] Facebook CTF. *Facebook CTF, Github.com.* [Online].

[17] Y. Cherdantseva and J. Hilton, *The Evolution of Information Security Goals from the 1960s to today.* Cardif University, 2012. [Online].

[18] "picoCTF". *picoCTF. https://picoctf.com/.* [Online].

[19] "What is picoCTF". *picoCTF. https://picoctf.com/about.* [Online].

[20] "OverTheWire". *Over the Wire. https://overthewire.org/wargames/.* [Online].

[21] "Workstation Player : Run a Second, Isolated Operating System on a Single PC with VMware Workstation Player", *VMware*, 2019. [Online].

[22] "Virtualization Technology & Virtual Machine Software: What is Virtualization?", *VMware*, 2019. [Online].

[23] "Network Connection Types: VMware Workstation 11 Documentation Center", *Pubs.vmware.com*, 2019. [Online].

[24] "VMware - Cloud, Mobility, Networking & Security Solutions", VMware, 2019. [Online].

[25] "The leading operating system for PCs, IoT devices, servers and the cloud — Ubuntu", *Ubuntu*, 2019. [Online].

[26] A. Verma, "Top 10 Best Linux Distros For 2018 - Ultimate Distro Choosing Guide", *Fossbytes*, 2018. [Online].

[27] S. Holmes, "Installing Ubuntu in VMware Player on Windows", *The Holmes Office.* [Online].

[28] "MySQL — The Most Popular Open-Source Database — Oracle", *Oracle.com*, 2019. [Online].

[29] "PRCCDC - Pacific Rim Collegiate Cyber Defense Competition", *Prccdc.org*, 2019. [Online].

[30] "National Collegiate Cyber Defense Competition", *Nationalccdc.org*, 2019. [Online].

[31] J. Ellingwood and M. Drake, *How to Instll Linux, Nginx, MySQL, PHP (LEMP stack) on Ubuntu 18.04*. Digital Ocean. 23-May, 2018. [Online].

[32] *Try VMware Workstation Player*. VMware. 2019. [Online].

[33] S. Jha, *Step by Step - Install VMware Workstation Player 15 in Windows 10*. Shaileshjha.com. 2019. [Online].

[34] *Ubuntu 16.04.6 LTS (Xenial Xerus)*. Canonical. 2018. [Online].

[35] V. Krishna, *How to Install Ubuntu in VMware Player [Windows]*. maketecheasier. 6-Jul, 2014. [Online].

[36] N. Lord, *What is the Principle of Least Privilege (POLP)? A Best Practice for Information Security and Compliance*. Digital Guardian: Data Insider. 12-Sep, 2018. [Online].

[37] *What is SQL?*. SQLCourse.com: Interactive Online SQL Training. 2019. [Online].

[38] *What is MySQL?*. Oracle: MySQL. 2019. [Online].

[39] *SQL Injection*. w3schools.com. 2019. [Online].

[40] *PHP Prepared Statements*. w3schools.com. 2019. [Online].

[41] *What is Virtualization?*. VMware. 2019. [Online].

[42] "Raytheon: Customer Success Is Our Mission," *raytheon.com*, 2019. [Online].

[43] "Faces of Cyber - Justin," *YouTube: Raytheon*, 2018.

[44] "Faces of Cyber - Mariah," *YouTube: Raytheon*, 2018.

[45] "Faces of Cyber - Hunter," *YouTube: Raytheon*, 2018.

[46] "University of Virginia's reign as Collegiate Cyber Champions continues," *YouTube: Raytheon*, 2019.

[47] "Digital Transformation — Dell Technologies," *delltechnologies.com*, 2019.

[48] "Oracle VM VirtualBox," *virtualbox.org*, 2019.

[49] A. Emmanoulopoulou, "Ubuntu is Everywhere!," *Ubuntu Blog*. Apr, 2016.

[50] "PostgreSQL: The World's Most Advanced Open Source Relational Database," *postgresql.org*, 2019.

[51] "The MariaDB Foundation - Supporting continuity and open collaboration in the MariaDB ecosystem," *mariadb.org*, 2019.

[52] "Switch To Static IP Address On Ubuntu 17.04—17.10," *Website for Students*. 2017.

[53] "Compare Workstation Player or Workstation Pro - Choose the Right Version for You," *vmware.com*, 2019.

[54] W. Newhouse, S. Keith, B. Scribner, and G. Witte, "NIST Special Publication 800-181: National Initiative for Cybersecurity Education (NICE) Cybersecurity Workforce Framework," *U.S. Department of Commerce, National Institute of Standards and Technology*. Aug 2017.

[55] "National Initiative for Cybersecurity Education," *U.S. Department of Commerce, National Institute of Standards and Technology*. 2019. [Online].

[56] "NICE Cybersecurity Workforce Framework," *U.S. Department of Homeland Security, National Initiative for Cybersecurity Careers and Studies*. May 2019. [Online].

[57] "Interactive NICE Framework Mapping," *SANS Institute*. 2018. [Online].

[58] "Incident Briefing (ICS 201)", *U.S. Federal Emergency Management Agency*. [Online].

[59] "National Cyber Incident Response Plan", *U.S. Department of Homeland Security*. Dec 2016.

[60] P. Cichonski, T. Milar, T. Grance, and K. Scarfone, "SP800-61 r2: Computer Security Incident Handling Guide", *U.S. Department of Commerce, National Institute of Standards and Technology*. Aug 2012.

[61] "US-CERT Incident Reporting System", *U.S. Department of Homeland Security, Cybersecurity and Infrastructure Security Agency*. [Online].

[62] E. Girken, "Incident Response Steps Comparison Guide for SANS and NIST", *AT&T Cybersecurity*. Jan 2019. [Online].

[63] J. Zilar, "Getting Started with your Contact Center", *Digital.gov*. Apr 2018. [Online].

[64] "Strategy 6Q: Standards for Customer Service", *U.S. Department of Health and Human Services, Agency for Healthcare Research and Quality*. Jan 2018. [Online].

[65] K. Lebing, "Improving Customer Service Through Effective Project Management", *U.S. Office of Personnel Management, Performance Management Practitioner Series*. Sep 1997.

[66] T.K. Connellan and R. Zemke, "Sustaining Knock Your Socks Off Service", *AMACOM, a division of American Management Association*. 1993. [Print].

[67] "Business Voicemail Greetings: 5 Sample Scripts", *onsip.com*. 2019. [Online].

[68] J. Silver and S. Spencer, "8 Small Business Voicemail Greeting Examples You Can Use Right Now", *Frontier Business*. [Online].

[69] R. Bozeman, "The Complete 2020 Guide to Professional Business Voicemail Greetings", *Startup Stockpile*. 2019. [Online].

[70] T. Pugielli, "Best Practices for Security Incident Reporting in Healthcare", *Turk-Key Technologies*. Oct 2018. [Online].

[71] N. Lord, "What is Security Incident Management? The Cybersecurity Incident Management Process, Examples, Best Practices, and More", *Digital Guardian's Data Insider*. Sep 2018. [Online].

[72] L. Voigt, "5 Best Practices for Your Incident Response Plan", *Exabeam*. Jul 2018. [Online].

[73] D. Bareja, "10 security incident mangement best practices", *Computer Weekly*. Jan 2011. [Online].

[74] M. Assouline, "4 Proven Strategies for Diffusing an Angry Customer", *Hub Spot*. Apr 2018. [Online].

[75] M. Solomon, "Turn Those Upset Customers Around: Best Practices For Customer Service Recovery", *Forbes*. Oct 2017. [Online].

[76] B. Metivier, "How to Detect and Respond to Insider Threats", *Sage Data Security*. Aug 2017. [Online].

[77] R. Felton, J. Hair, A. Grahn, A. Kizziah, and D. O'Leary, "5 Keys to Addressing Insider Threats", *Sirius Edge*. Jan 2019. [Online].

[78] T. Eck and A. Grahn, "7 Steps to Effective Data Classification", *Sirius Edge*. Aug 2019. [Online].

[79] B. Johnson, "Data Classification Policy Best Practices — A Practical Guide to SOC 2 Compliance", *Strong DM*. Nov 2018. [Online].

[80] A. McCreath, "Data classification best practices create effective policy", *Computer Weekly*. Apr 2009. [Online].

[81] S. Hillenburg, "SpongeBob SquarePants", *United Plankton Pictures, Nickelodeon Animation Studios, Viacom Media Networks*. 1999-2012. [Television].

[82] "Guidelines for Effective Briefings", *Massachusetts Institute of Technology*. [Online].

[83] J. Lohrey, "How to Give a Successful Briefing", *Chron*. Sep 2019. [Online].

[84] "Know the types of cyber threats", *mass.gov*, 2019. [Online].

[85] "Cyber Threat Source Descriptions", *U.S. Department of Homeland Security, Cybersecurity and Infrastructure Security Agency*. [Online].

[86] "NICE Challenge Project", *National Cyber Range*. [Online].

[87] "NICE Summer 2019 eNewsletter", *U.S. Department of Commerce, NIST*. 2019. [Online].

[88] "NICE Fall 2019 eNewsleter", *U.S. Department of Commerce, NIST*. 2019. [Online].

[89] "4 Types of Organizational Structures", *Point Park University*. Feb 2018. [Online].

[90] "Types of Organizational Charts and How to Use Them", *SmartDraw*. [Online].

[91] A. Hayes, "10 Tips for Choosing Effective Team Members", *AboutLeaders*. May 2018. [Online].

[92] T. Jones, "6 Tips for Choosing Effective Project Team Members", *BrightWork*. Dec 2017. [Online].

[93] A.A. Jillepalli, D. Conte de Leon, F.T. Sheldon, and M.A. Haney, "Enterprise-level Hardening of Web Browsers for Microsoft Windows", *International Journal of Computing and Digital Systems, Sys. 7, No. 5 (Sep-2018)*. Sep 2018.

[94] B. Mathers, J. Flores, L. Poggemeyer, Y. Shengjin, S. Kumar, E. Ross, and A. Rechenberg, "Securing Domain Controllers Against Attack", *Microsoft*. Jun 2017. [Online].

[95] B. Mathers, J. Flores, K. Chewie, Y. Shengjin, S. Kumar, E. Ross, K. Nikolaev, L. Poggemeyer, and C. Watson, "Implementing Least-Privilege Administrative Models", *Microsoft*. Aug 2018. [Online].

[96] M. Miller, "Active Directory Security Explained & 7 Best Practices", *BeyondTrust*. Nov 2018. [Online].

[97] "Tabletop Exercises: Sex Scenarios to Help Prepare Your Cybersecurity Team", *Center for Internet Security*. Oct 2018. [Online].

[98] S. Ewing, "4 Cyber Incident Scenarios You Should Exercise and Test", *Delta Risk*. Sep 2016. [Online].

[99] J. Kick, "Cyber Exercise Playbook", *MITRE*. Appendix A. Nov 2014.

[100] "SEC503: Intrusion Detection In-Depth", *SANS*. [Training].

[101] "FOR508: Advanced Incident Response, Threat Hunting, and Digital Forensics", *SANS*. [Training].

[102] "MGT512: Security Leadership Essentials For Managers", *SANS*. [Training].

[103] "Udemy: Learn on Your Schedule", *Udemy*. [Webpage].

[104] "ISO/IEC 27001 Information Security Management", *International Organization for Standardization*. [Online].

[105] A.J. Segovia, "ISO 27001 vs. ISO 27032", *Advisera.com*. Aug 2015. [Online].

[106] L. Irwin, "What is the ISO 27000 series of standards?", *IT Governance*. Oct 2019. [Online].

[107] "Defense Federal Acquisition Regulation", *U.S. Department of Defense*. Oct 2019. [Online].

[108] "Defense Federal Acquisition Regulation Supplement (DFARS)", *Federal Register, National Archives*. Nov 2019. [Online].

[109] "Secure Controls Framework", *Secure Controls Framework (SCF)*. 2019. [Online].

[110] "Security & Privacy Metaframework", *Secure Controls Framework (SCF)*. 2019. [Online].

[111] T. Cornelius, "Launch of the Secure Controls Framework", *PRWeb CISION*. Feb 2018. [Online].

- **Incident Response (CIR)** [54, 55, 56, 57]

  "Responds to crises or urgent situations within the pertinent domain to mitigate immediate and potential threats. Uses mitigation, preparedness, and response and recovery approaches, as needed, to maximize survival of life, preservation of property, and information security. Investigates and analyzes all relevant response activities."

  PR-CIR-001: Cyber Defense Incident Responder Work Role Description

  "Investigates, analyzes, and responds to cyber incidents within the network environment or enclave."

- **Cybersecurity Defense Infrastructure Support (INF)** [54, 55, 56, 57]

  "Tests, implements, deploys, maintains, reviews, and administers the infrastructure hardware and software that are required to effectively manage the computer network defense service provider network and resources. Monitors network to actively remediate unauthorized activities."

  PR-INF-001: Cyber Defense Infrastructure Support Specialist Work Role Description

  "Tests, implements, deploys, maintains, and administers the infrastructure hardware and software."

- **Data Administration (DTA)** [54, 55, 56, 57]

  "Develops and administers databases and/or data management systems that allow for the storage, query, protection, and utilization of data."

  OM-DTA-001: Database Administrator Work Role Description

  "Administers databases and/or data management systems that allow for the secure storage, query, protection, and utilization of data."

- **Customer Service and Technical Support (STS)** [54, 55, 56, 57]

  "Addresses problems; installs, configures, troubleshoots, and provides maintenance and training in response to customer requirements or inquiries (e.g., tiered-level customer sup-

port). Typically provides initial incident information to the Incident Response (IR) Specialty."

OM-STS-001: Technical Support Specialist Work Role Description

"Provides technical support to customers who need assistance utilizing client-level hardware and software in accordance with established or approved organizational process components (i.e., Master Incident Management Plan, when applicable)."

- **Software Development (DEV)** [54, 55, 56, 57]

  "Develops and writes/codes new (or modifies existing) computer applications, software, or specialized utility programs following software assurance best practices."

  SP-DEV-001: Software Developer Work Role Description

  "Develops, creates, maintains, and writes/codes new (or modifies existing) computer applications, software, or specialized utility programs."

  SP-DEV-002: Secure Software Assessor Work Role Description [54, 55, 56, 57]

  "Analyzes the security of new or existing computer applications, software, or specialized utility programs and provides actionable results."

- **Technology R&D (TRD)** [54, 55, 56, 57]

  "Conducts technology assessment and integration processes; provides and supports a prototype capability and/or evaluates its utility."

  SP-TRD-001: Research & Development Specialist Work Role Description

  "Conducts software and systems engineering and software systems research to develop new capabilities, ensuring cybersecurity is fully integrated. Conducts comprehensive technology research to evaluate potential vulnerabilities in cyberspace systems."

- **Test and Evaluation (TST)** [54, 55, 56, 57]

  "Develops and conducts tests of systems to evaluate compliance with specifications and requirements by applying principles and methods for cost-effective planning, evaluating, verifying, and validating of technical, functional, and performance characteristics (including interoperability) of systems or elements of systems incorporating IT."

  SP-TST-001: System Testing and Evaluation Specialist Work Role Description

"Plans, prepares, and executes tests of systems to evaluate results against specifications and requirements as well as analyze/report test results."