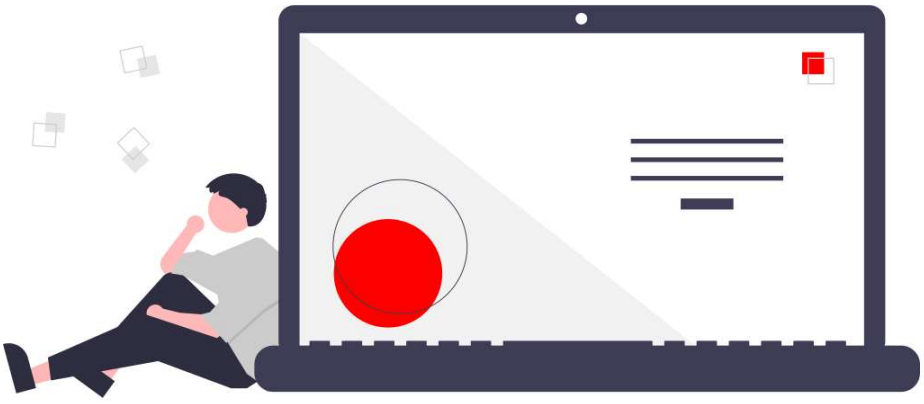
# Secure IDE for Red Teaming



**COMPX576 – Programming project**

**Venkatasubramanian Sankaranarayanan**

**Student ID: 1649356**

# Table of contents

## Project proposal

### *Introduction*

We all know that there are Integrated Development Environment customized testing environment for Software Development. What about Security testing? Here I have proposed a secure environment for practicing various attacking skills for Red teaming with few challenges. It can help attackers enhance their skills in various attacking methodologies inside a secure environment without disturbing the real world devices.

### *Methodology*

I am including the methodologies, which include developing the environment, customization, deployment and testing.

1. Choosing suitable Operating system for the room (environment)
2. Setting up server
3. Installation of LAMP stack architecture
4. Building CMS (Content Management System) on top of LAMP stack
5. Creating custom plugin for CMS
6. Implementing Access management to make privilege escalation attack more challenging
7. Creating and hiding relevant ‘flags’ as a part of the challenge
8. Deployment and testing
9. Final product submission

### *Requirements*

1. Hypervisor – Oracle VirtualBox
2. Operating System – Debian Based architecture
3. Attacking machine (another os in the hypervisor within the NAT network)

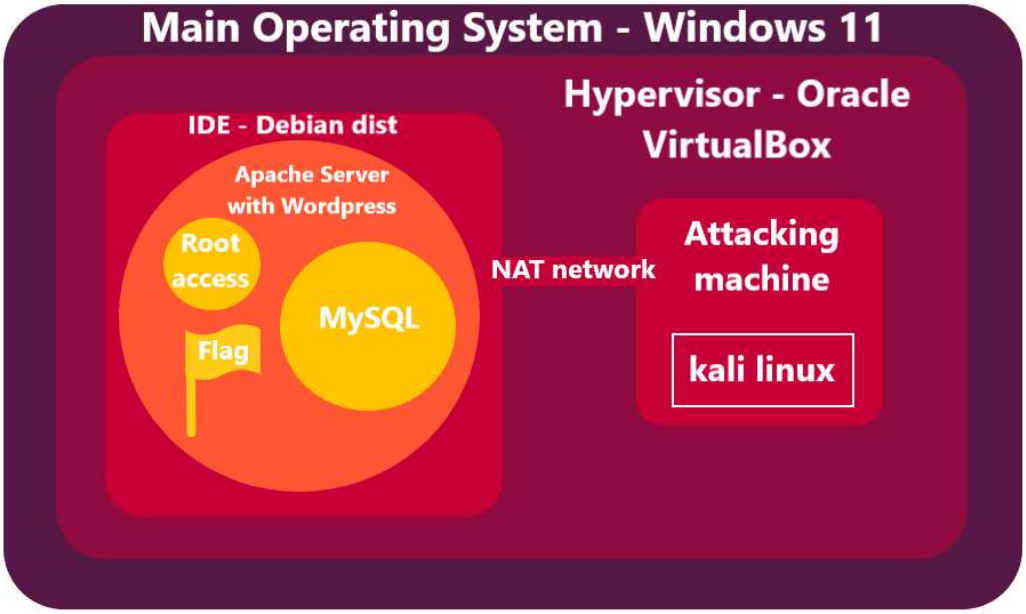
### *Technologies*

1. Hypervisor – Oracle VirtualBox
2. Operating System (Debian based OS)
3. Webserver – Apache/Ubuntu
4. Database – MySQL
5. Scripting Language – PHP/Bash
6. Content management System – Wordpress
7. Secure Shell with RSA/passwd login
8. Networking - TCP/IP and NAT
9. IDE (Code) – nano/Vim

### *What I am learning for this project?*

1. Server configuration
2. Custom CMS plugin development
3. PHP – Server side programming language
4. Access level definition – Custom Linux automations

### *Architecture*



## *Conclusion*

This project helps in Cybersecurity training bridging the gap between theoretical and practical implementations especially designed for the Red team. This allows people who have keen interest on red teaming and are ready to take up tough challenges without disturbing the real world entities. Ultimately, it contributes to the Cybersecurity community who are ethically bound to the compliances.

## Week – 2

### *Weekly target*

1. Choosing apt Linux distribution
2. Installation of Server and it's configuration
3. Implementation of LAMP Stack over the server
4. Prepare the internal network (Network Address translation and port forwarding)

### *Challenges*

1. Initially I chose Kubuntu for Base Operating system. I noticed that it consumes a lot of graphics for virtualization because of KDE-Plasma setup. The environment crashed repeatedly. Therefore, I chose to go with "A platform which consumes less graphics and allows user to have Super user permission in ease". I chose "kali Linux" which uses GNOME-2, which consumes less graphics.
2. After the installation of Kali Linux, I started to install "Ubuntu Server" where the installation of CMS is going to take place. But the server crashed at first due to misconfiguration issues. Then I re-installed that and started executing the CMS. I was implementing the server installation and setup every time whenever I need to develop and test the plugin code. So, I decided to install "Docker" container.
3. After the installation of Docker, I wrote a custom "yml" file for the docker build. I got errors due to version mismatch. After going through the errors, I noticed that, I have installed V3 of "docker-compose" to build the instance. I downgraded the mentioned service to V2 to create and run the instances.

## *Conclusion*

Though I faced many challenges, I managed to complete this week's tasks. It helped me to learn a new technology "Docker" for container management. The challenges I faced helped me to get to know about how important version of software is.

## *Week 3-4 Target*

Creating custom Plugin for CMS (Wordpress)

## *References (Week-2)*


1. https://appsecexplained.gitbook.io/appsecexplained/scripts/docker-compose.yml-files/wordpress - Docker instance creation for WordPress and Database connection
2. Few YouTube videos on tutorials to install Ubuntu server, Docker and fixing errors.

## Week – 3

### *Weekly target*

1. To create an user flag and root flag which must be captured by the attacker

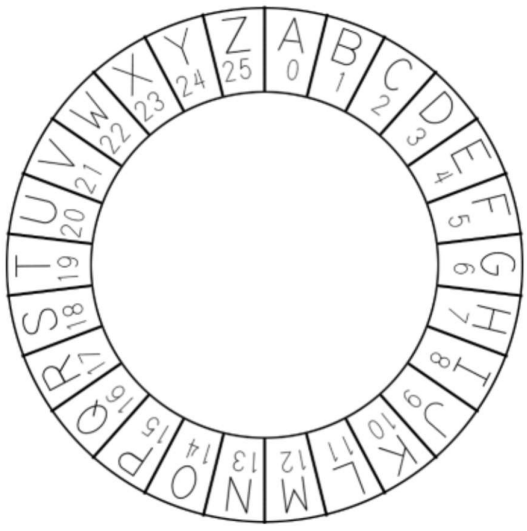
### *Flag Specification*

1. The flag must be encrypted with any algorithm or cipher
2. We must provide hint to decrypt the cipher and algorithm
3. Both local user and root user must have separate flags

### *Local user flag creation*

For Local user, we are implementing ROT13 cipher. For hint, we are going to name the text file which contains user flag as ’13.txt’

| | |
|---|---|
| **Encryption** | **C = O + 13** |
| **Decryption** | **O = C - 13** |




**Credit -** https://www.researchgate.net/figure/Circular-positional-alphabet-and-position-values_fig1_330521841

### *Root user flag creation*

For Root user, we are implementing Vigenère cipher. For hint, we are going to name the text file, which contains root flag as 'le_chiffrage_indéchiffrable.txt'

This encryption method requires a key to decrypt the cipher. So I am going to put a hint below the encryption as 'userflag'.

| | |
|---|---|
| **Encryption** | **C_i = (P_i + K_i) mod 26** |
| **Decryption** | **P_i = (C_i - K_i + 26) mod 26** |

### *Challenges faced*

1. Understanding and implementing the cipher and encryption techniques. Written code for the above mentioned encryptions.

### *Conclusion*

Initially, I was about to complete the CMS plugin creation. Then I decided to create flags first so that it can be with uploaded once the plugin is created. We have few more challenges to face during the development.

## Week 4

### *Weekly Target*

1. To write a custom Wordpress plugin with php which can be run over docker container with Ubuntu
2. Test whether the plugin runs and gets an input from the user

### *Plugin Specs*

1. Take input from the user
2. Can take “any” input from the user
    a. This is a processed threat that needs to be exploited by the attacker.
    b. This is termed as “Command Injection Vulnerability”

### *Code*

```
function report_reader_include_file($atts) {
//user logged in
if (!current_user_can('manage_options')) {
return 'You do not have sufficient permissions to access this content.';
}
// get shortcode params
extract(shortcode_atts(array(
'path' -> '',
), $atts));
// $path = sanitize_text_field($path);
// construct the full path
$full_path = ABSPATH . $path;
if (!file_exists($full_path)) {
        return 'The specified file does not exist.';
}
// return the file contents
return file_get_contents($full_path);
add_shortcode('include_report',report_reader_include_file');

//Code needs be sanitized more and few bugs needs to be fixed.
```