Penetration Test Report for Kioptrix Level 3

example@example.com

OSID: XXXXX

**Table of Contents**

[1.0 High Level Summary 3](#_Toc30167503)

[1.1 Recommendations 3](#_Toc30167504)

[2.0 Methodologies 4](#_Toc30167505)

[2.1 Information Gathering 4](#_Toc30167506)

[2.2 Penetration 4](#_Toc30167507)

[System IP: 192.168.1.14 4](#_Toc30167508)

[Service Enumeration 4](#_Toc30167509)

[Privilege Escalation 5](#_Toc30167510)

[2.3 House Cleaning 6](#_Toc30167511)

# 1.0 High Level Summary

I was tasked with performing an internal penetration test towards Kioptrix Lab. An internal penetration test is a dedicated attack against internally connected systems. The focus of this test is to perform attacks, similar to those of a hacker and attempt to infiltrate Kioptrix Lab systems – the THINC.local domain. My overall objective was to evaluate the network, identify systems, and exploit flaws while reporting the findings to the web.

When performing the internal penetration test, there were several alarming vulnerabilities that were identified on Kioptrix’s network. When performing the attacks, I was able to gain access to multiple machines, primarily due to outdated patches and poor security configurations. During the testing, I had administrative level access to multiple systems. All systems were successfully exploited and access granted. These systems as well as a brief description on how access was obtained are listed below:

* 192.168.1.14 – SQL Injection on the Gallery Page

## 1.1 Recommendations

I recommend patching the vulnerabilities identified during the testing to ensure that an attacker cannot exploit these systems in the future. One thing to remember is that these systems require frequent patching and once patched, should remain on a regular patch program to protect additional vulnerabilities that are discovered at a later date.

# 2.0 Methodologies

I utilized a widely adopted approach to performing penetration testing that is effective in testing how well the Kioptrix Lab environment is secured. Below is a breakdown of how I was able to identify and exploit the variety of systems and includes all individual vulnerabilities found.

## 2.1 Information Gathering

The information gathering portion of a penetration test focuses on identifying the scope of the penetration test. During this penetration test, I was tasked with exploiting the Lab network. The specific IP addresses were:

* 192.168.1.14

## 2.2 Penetration

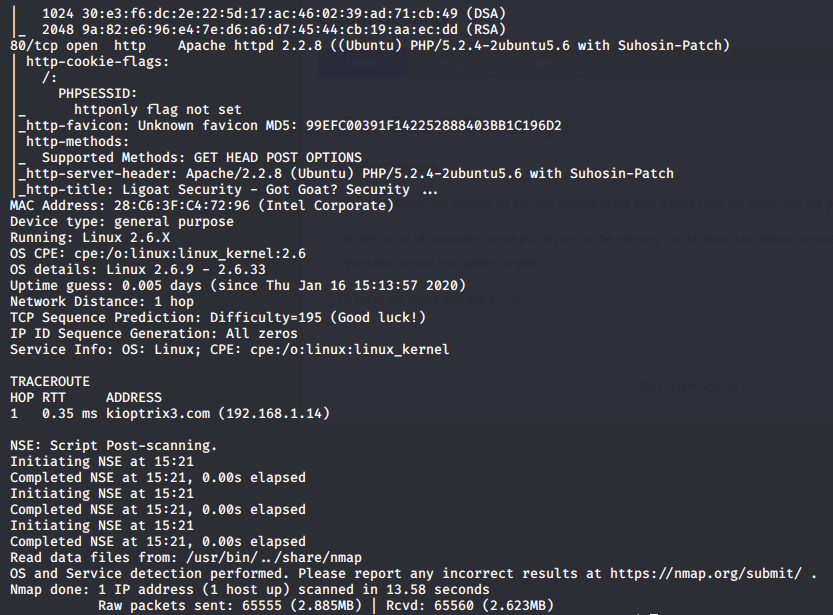
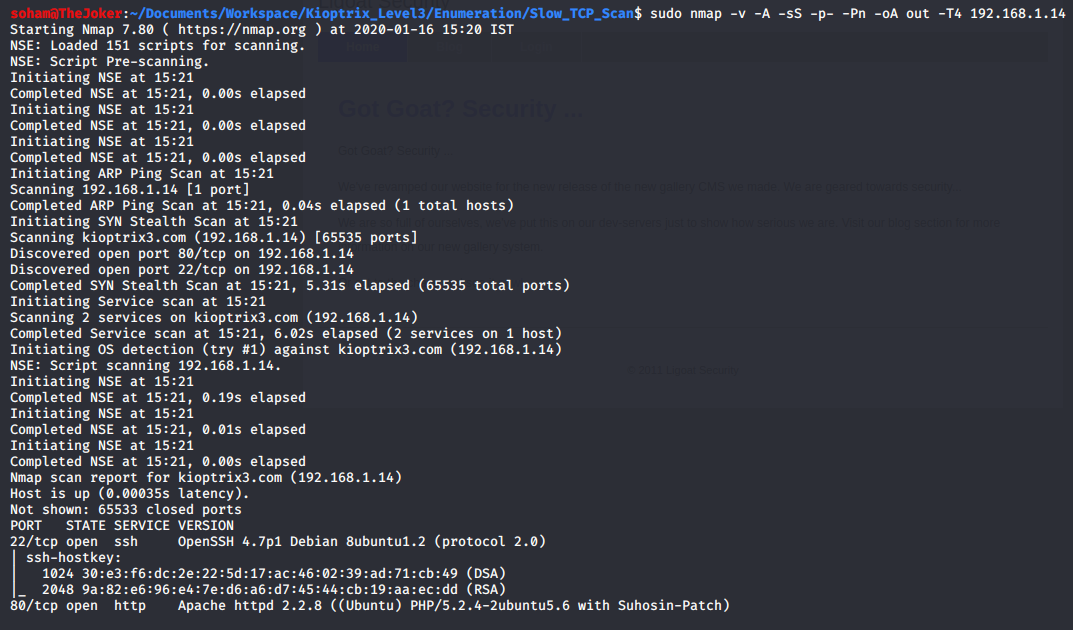
The penetration testing portions of the assessment focus heavily on gaining access to a variety of systems. During this penetration test, I was able to successfully gain access to 1 out of the 1 system(s).

## System IP: 192.168.1.14

#### Service Enumeration

|  |  |
| --- | --- |
| Server IP Address | Ports Open |
| 192.168.1.14 | TCP: 22, 80 |

Nmap Scan Result:

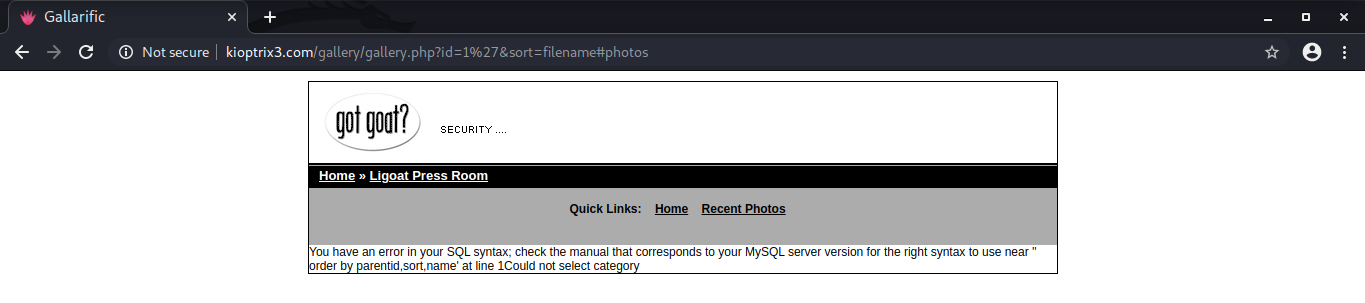


**SQL Injection on Gallery Page on Port 80**

Since port 80 is open, the webpage was visited using a web browser.

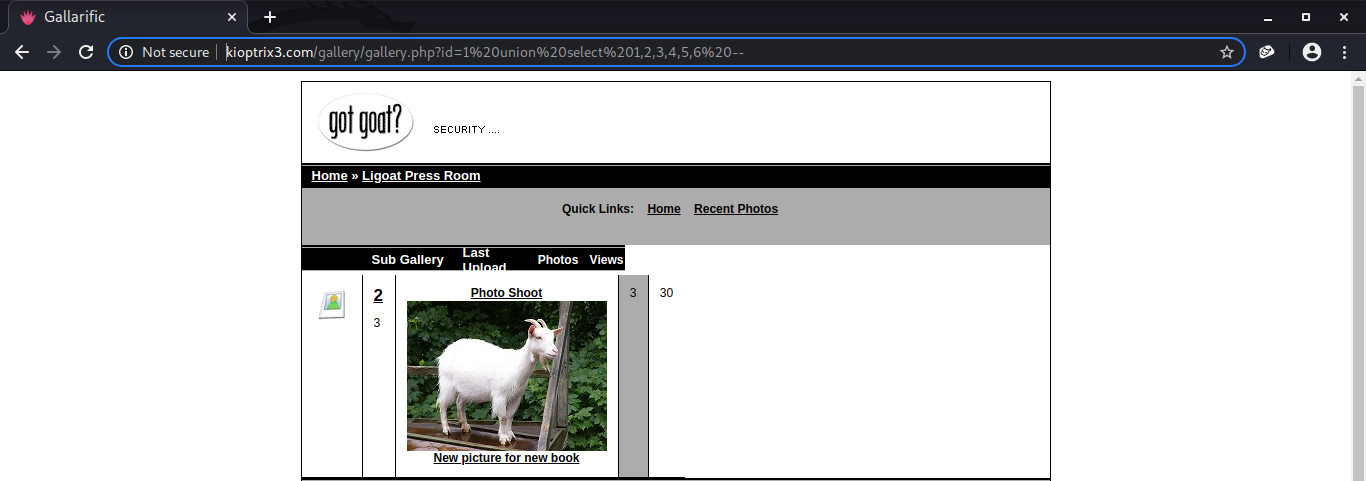
The links were checked out by manually web crawling. On the gallery page (kioptrix3.com/gallery/gallery.php), it was discovered that adding an ‘ in the id parameter, and then visiting <http://kioptrix.com/gallery/gallery.php?id=1>’ would show a sql error message. This is often an indication of presence of sql injection vulnerability.

*Sql error message:*

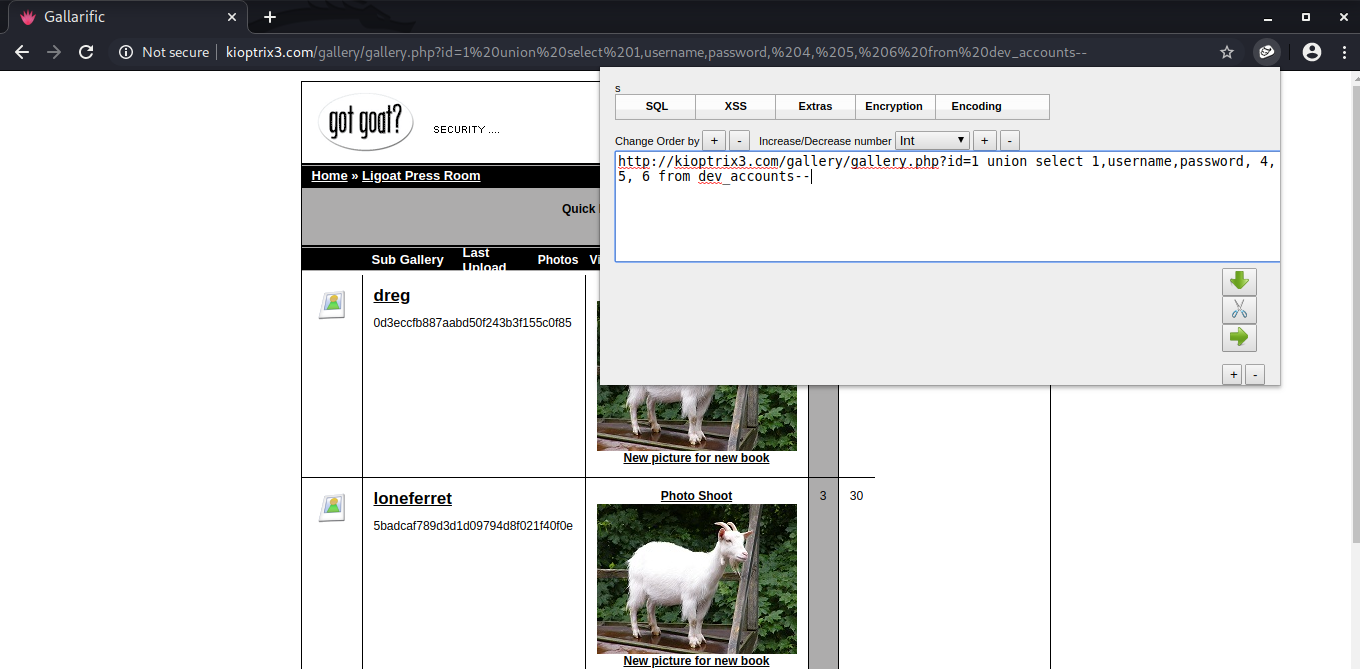
**

On further exploration it was concluded that there was a sql injection vulnerability present, and it was used to get user data which included usernames, and password hashes.

*Exploitation of the vulnerability:*

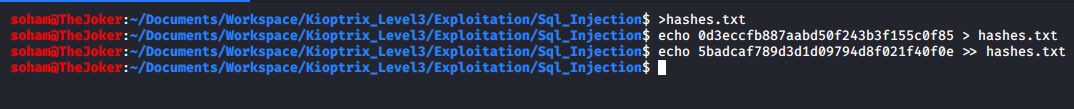
**

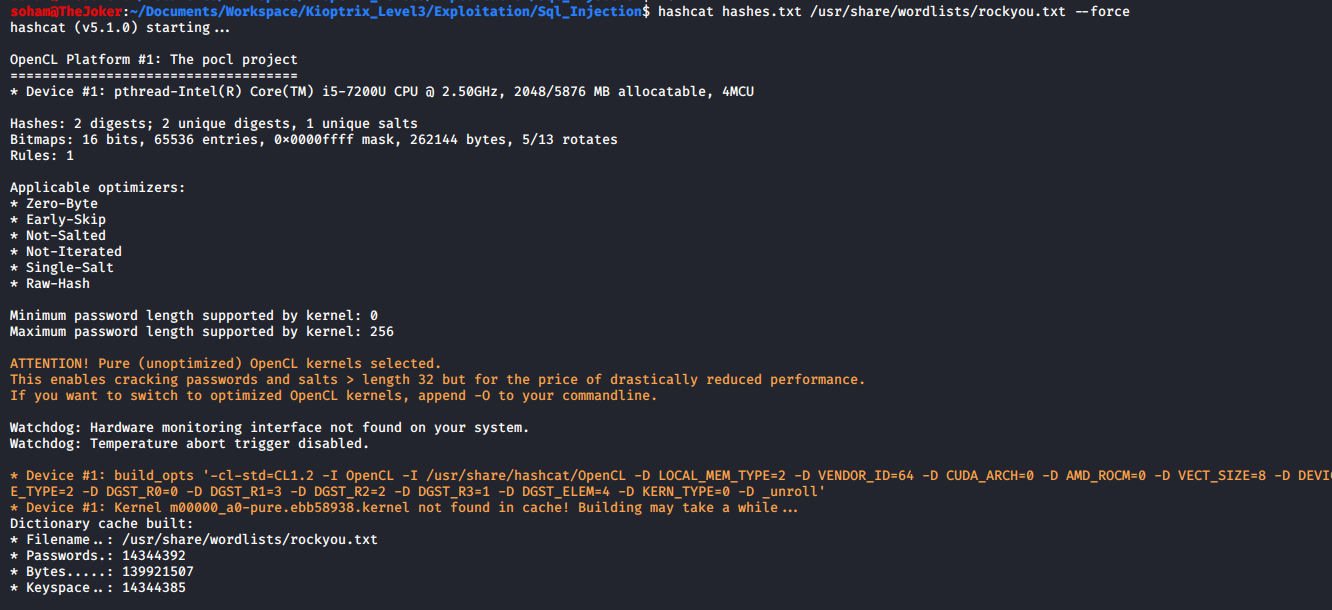
*User data*

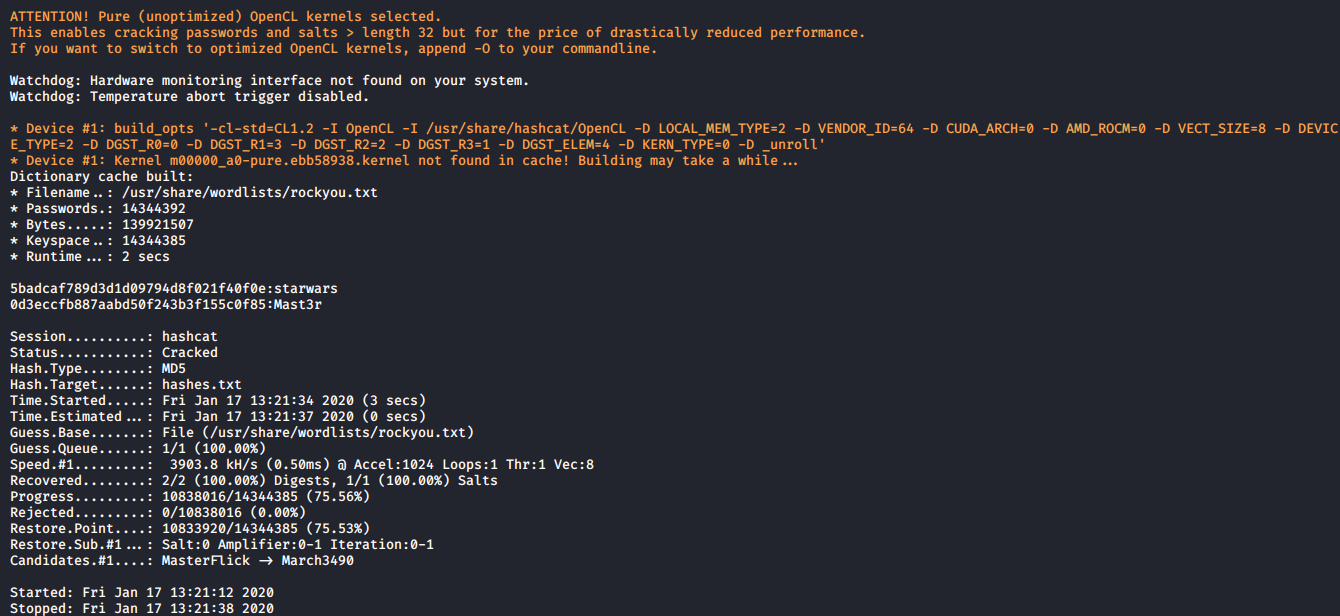
**

These password hashes were cracked using hashcat.

*Hashcat:*

**

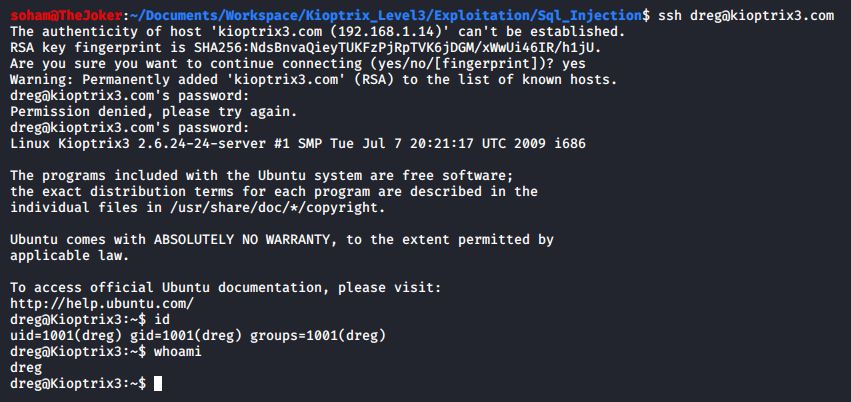
**

**

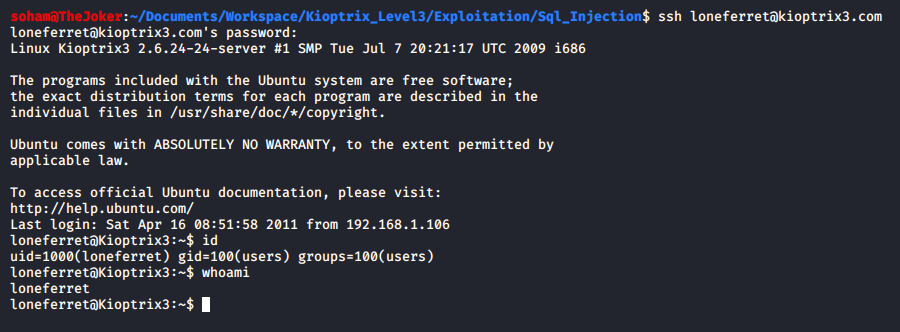
*C:\Users\ADMIN\Desktop\Temp\SQLi\Screenshot_2020-01-17_13-23-01.png*

The credentials were then used to ssh into the system there by getting a shell on the system.

*Ssh with dreg:*

**

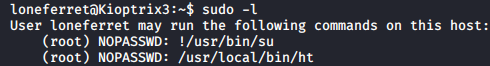
*Ssh with loneferret:*

**

#### Privilege Escalation

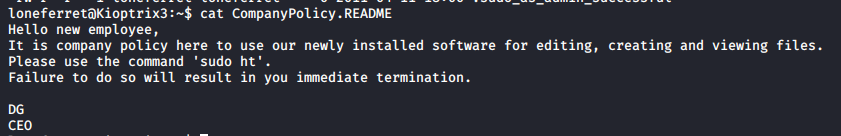
With a bit of enumeration, it was discovered that the user loneferret could execute certain commands as root user thereby giving an opening for privilege escalation. One of the commands that could be executed as root was ‘ht’.

*sudo –l*

**

CompanyPolicy.README file gives the information that ht is a text editor.

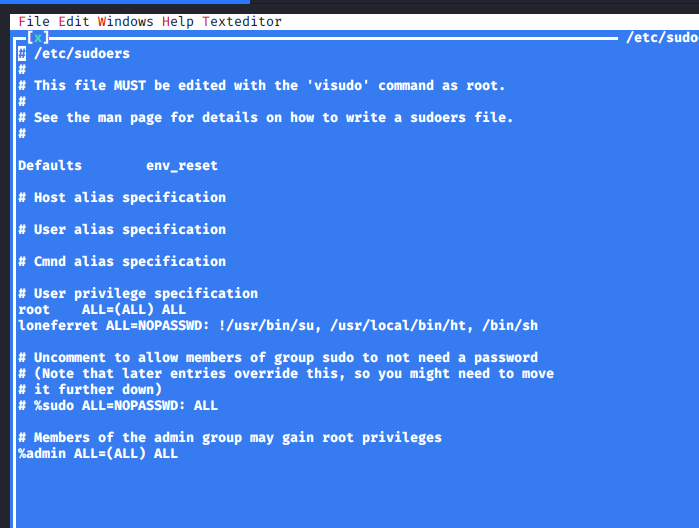
*companypolicy.readme*

**

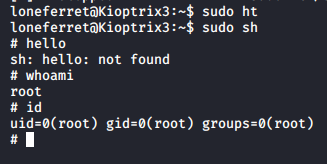
These pieces of information implied that loneferret could read, and write to any file since ht was being executed as root user.

Therefore, the /etc/sudoers was edited to give loneferret the permission to executed /bin/sh thereby allowing for arbitrary code execution as root.

*Edited sudoers file:*

**

*Sh as root user:*

**

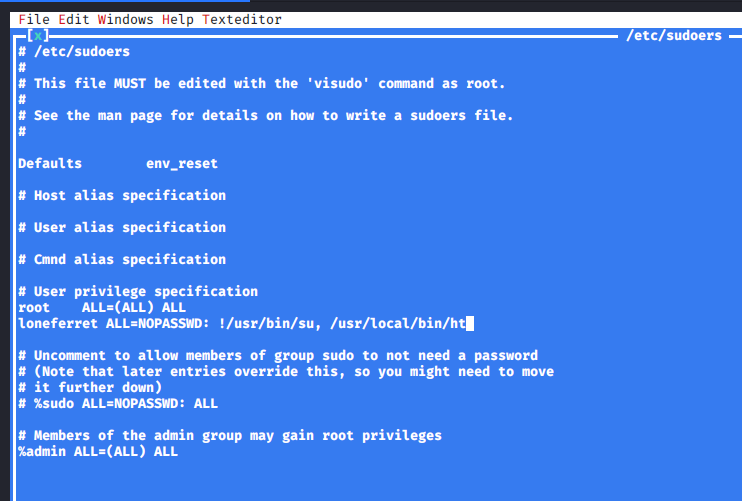
Hence, the full compromise of the system was completed.

## 2.3 House Cleaning

The house cleaning portions of the assessment ensures that remnants of the penetration test are removed. Often fragments of tools or user accounts are left on an organization's computer which can cause security issues down the road. Ensuring that we are meticulous and no remnants of our penetration test are left over is important.

There was only one change made in the system: adding of “/bin/sh” to /etc/sudoers file. After the colleting trophies from the system, the original /etc/sudoers file was restored.

*Original sudoers file*

**