LAB 13
Jonathon Schnell
CPRE 230

Secure the Box

Identify the vulnerability	What harm could it do?	How to fix it

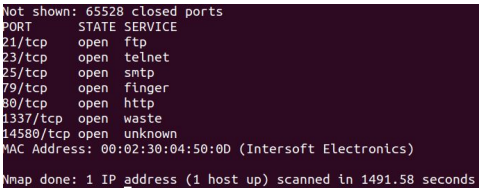
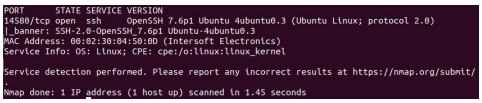
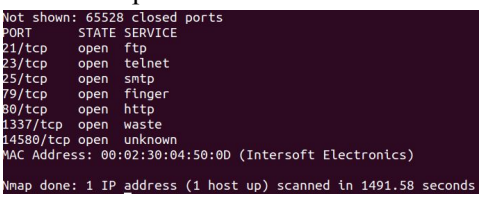
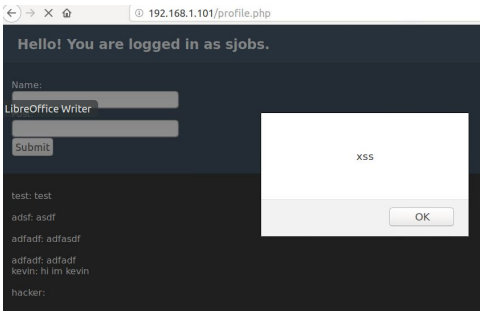
1. Misconfigured file permissions
`root@capstone:/srv/ftp# ls -la`
`total 36`
`drwxr-xr-x  7 root ftp  4096 Aug 17 22:35 .`
`drwxr-xr-x  3 root root  4096 Oct 14  2017 ..`
`drwxr-xr-x 102 root root 12288 Aug 17 22:35 fun_files`
`drwxrwxrwx  3 root root  4096 Oct 14  2017 hacks_here`
`drwxrwxrwx  2 root root  4096 Oct 14  2017 irc_chat_here_too`
`drwxrwxrwx  2 root root  4096 Oct 14  2017 vulns_here`
`drwxr-xr-x 23 root root  4096 Aug 17 21:31 whats_in_here`
	any script run in these folders is run as a root but anyone can access, edit, and execute the folder and files within the folder. This could easily be used to escalate a user's privileges or put a backdoor in place.	The fix to this security flaw would be to determine who needs access to this folder and at the very least prevent everyone from writing to this folder and executing this folder.
		>chmod 770
		or
		>chmod 750

2.Rouge user accounts
`messagebus:*:17394:0:99999:7:::`
`uuidd:*:17394:0:99999:7:::`
`dnsmasq:*:17394:0:99999:7:::`
`sgrpe230:$6$jKoSUO2N$19.jDuZU305zPFjiTpAkdPL8mU78o82911.UZUAV1uN5112sPLmCyDMwL.j6n01EGJV11Tn2fAhnS5cBu`
`hiltP0:17394:0:99999:7:::`
`tom:$6$49gxU.9MH$dOEPnJFU9r1XAg2FUxGub4MzSO.nFhx5vWED7pr9NYhW01kC6o296.ntPiwe8etNMtAKYYFXbaYghRMY.jF2U`
`00:17453:0:99999:7:::`
`tooe:$6$qhuP7imh$a1440mhAG1enD7chp1aowf2MJvkRV.3erK95lJvcUX35kgw2YnJSy2qVoq4bajj07UowLjJf94Cldg4fU3j`
`gR/:17453:0:99999:7:::`
`jrg:$6$uBgi3mBaS$UFKmckAJSjib7H23GWgBNN57VgUhhdgoxW/omnMtKBFsh9Kiw2P54UpPwfgCaWbuLbgpS8M7P4iNCu1DWUh`
`hi:17453:0:99999:7:::`
`backdoor::17453:0:99999:7:::`
`bob:$6$tXUpxuwQ$1pIxd65OrE2f4zQt6/dpQO629rCfIE1fsH2VQ3.MaICt4Wgpu01E41Hv2KFasH9eiDtPFQ0C6RDgATBYMUl`
`l/:17453:0:99999:7:::`
`alice:$6$tSP.cdGU$CA.J5R5i9B1TV.iX7PlQK7T1A2/0.j4j/0f.bXgINM4llePHqAdMaaJ/Z2n.d4r4uCzPSibCqkJgmXPrfUb`
`D7w:17453:0:99999:7:::`
`alex:$6$Re0Dr9qg$k9vuI7v05SUfqaw25AuxhQ/g0ot76YCkF051LMjTt.jMcxbDhn395Ch23bLHobrg5c7Sg5Iu9rEZEeOMYxQ`
`iP1:17453:0:99999:7:::`
`eve:$6$YrQdeRCT$2XnIfw.nBfHGh2dEJLEg0xUpq7sz40514qN6uAjiFDdM9zZkdIcWAB3f62BUoZLg8PQuz1fE1B.ZSmxZK1Yb`
`00:17453:0:99999:7:::`
`sshd:*:17453:0:99999:7:::`
`telnetd:*:17453:0:99999:7:::`
`mongodb:*:17453:0:99999:7:::`
`ftp:*:17453:0:99999:7:::`
`mysql:!:17453:0:99999:7:::`
`postfix:*:17453:0:99999:7:::`
`landscape:*:18126:0:99999:7:::`
`pollinate:*:18126:0:99999:7:::`
	In the etc shadow file we see the users and their password hashes. Two blatant flaws are the ex-employee bob still having a user on this box and a backdoor account with no password.	The fix would be to remove bob's user as well as the backdoor user from the machine.
		I would audit the whole list based on active employees and have all employees change their password immediately on next login.

3.firewall configuration
`root@capstone:/# sudo ufw status`
`Status: inactive`
	While this is an internal box and there is a firewall protecting it from internet traffic it is still a good practice to have a local firewall protecting the box from anyone with access to internal network or physical access to the building.	Enabling UFW and only allow required services such as http smtp and sftp.
		>sudo ufw enable
		>sudo ufw allow udp/22
		ect.

4.Unnecessary services running
`root@capstone:/var/www/html# mongo`
`MongoDB shell version v3.6.3`
`connecting to: mongodb://127.0.0.1:27017`
`MongoDB server version: 3.6.3`
`Server has startup warnings:`
`2019-11-21T23:36:30.809-0600 I CONTROL  [initandlisten]`
`2019-11-21T23:36:30.809-0600 I CONTROL  [initandlisten] ** WARNING: Access control is not enabled fo`
`r the database.`
`2019-11-21T23:36:30.809-0600 I CONTROL  [initandlisten] **  Read and write access to data an`
`d configuration is unrestricted.`
`2019-11-21T23:36:30.809-0600 I CONTROL  [initandlisten]`
`> db`
`test`
`> use test`
`switched to db test`
`> show collections`
`>`
	From the screenshot we can see that there is no password on and nothing in the mongo 'test' database it is likely that this a legacy or experimental service that was overlooked before deployment.	Disable mongodb on startup if it is not required. Or secure it with a password.

5.UPDATE/UPGRADE root@capstone:/# sudo apt-get update -v apt 1.6.11 (amd64) Supported modules: *Ver: Standard .deb *Pkg: Debian dpkg interface (Priority 30) Pkg: Debian APT solver interface (Priority -1000) Pkg: Debian APT planner interface (Priority -1000) S.L: 'deb' Debian binary tree S.L: 'deb-src' Debian source tree Idx: Debian Source Index Idx: Debian Package Index Idx: Debian Translation Index Idx: Debian dpkg status file Idx: Debian deb file Idx: Debian dsc file Idx: Debian control file Idx: EDSP scenario file Idx: EIPP scenario file	Keeping systems and services up to date is a huge security concern as old versions of software could possibly contain security vulnerabilities.	Update and upgrade the server on a regular consistent basis to avoid running legacy software that could potentially contain security flaws. It is important to note that an update is not just going to fix all of your security flaws. If an exploit is found and 'patched' the patch may require the administrator to reconfigure the software to utilize the new more secure features.
6.Running services as root -rw-r--r-- 1 root root 103 Oct 14 2017 vars.inc root@capstone:/var/www/html# cat vars.inc <?php $db_host = "localhost"; $db_name = "login_info"; $db_user = "root"; $db_password = "cpre230"; ?> root@capstone:/var/www/html# _ cpre230@capstone:/etc/ssh$ mysql -u root -p Enter password: Welcome to the MySQL monitor. Commands end with ; or \g. mysql> show tables; +----------+ | Tables_in_login_info | +----------+ | UsernamePassword | +----------+ 1 row in set (0.00 sec) mysql> SELECT * FROM UsernamePassword; +----------+----------+----------+ | usernameID | username | password | +----------+----------+----------+ | 1 | sjobs | hunter2 | | 2 | bgates | Cthulu15b@e | | 3 | djacobson | h@ckTh3Pl@net! | | 4 | sjackson | snakes? | | 5 | bmadoff | $$billy'all | +----------+----------+----------+ 5 rows in set (0.01 sec) mysql> _	Any remote service shouldnt be run as root because if there are any remotely exploitable vulnerabilities in the application the attacker can more easily obtain root privileges. Bad file perms leave plain text mysql passwords readable to any user. Using this password an attacker can read all the tables in the database and discover plain text login information for the http message board.	Create a user that does not have root privileges to host services such as apache and mysql.
7.Security by obscurity root@capstone:/var/www/html# netstat -plant Active Internet connections (servers and established) Proto Recv-Q Send-Q Local Address Foreign Address State PID/Program name tcp 0 0 127.0.0.1:3306 0.0.0.0:* LISTEN 1113/mysqld tcp 0 0 0.0.0.0:14500 0.0.0.0:* LISTEN 1457/sshd tcp 0 0 127.0.0.53:53 0.0.0.0:* LISTEN 659/systemd-resolve tcp 0 0 0.0.0.0:25 0.0.0.0:* LISTEN 1413/master tcp 0 0 0.0.0.0:1337 0.0.0.0:* LISTEN 974/nc tcp 0 0 127.0.0.1:27017 0.0.0.0:* LISTEN 928/mongod tcp 0 0 127.0.0.1:56036 127.0.0.1:27017 TIME_WAIT - tcp6 0 0 :::79 :::* LISTEN 1085/xinetd tcp6 0 0 :::80 :::* LISTEN 1147/apache2 tcp6 0 0 :::14500 :::* LISTEN 1457/sshd tcp6 0 0 :::21 :::* LISTEN 970/vsftpd tcp6 0 0 :::23 :::* LISTEN 1085/xinetd tcp6 0 0 :::25 :::* LISTEN 1413/master root@capstone:/var/www/html# _	The National Institute of Standards and Technology says about security by obscurity "System security should not depend on the secrecy of the implementation or its components." by putting ssh on a seemingly random port	To secure the ssh service on this box I would move it to the standard port 22. And generate new private public key pairs for each user that should be able to ssh into this box.

 	all we aren't really doing anything because an attacker can probe the port for a banner and discover what service is running on that mystery port. Although strategy is important to defending computer systems there is no advantage to be had here.	
8.Plaintext protocols 	Plaintext protocols such as Telnet, smtp, http, Ftp. plain text protocols can do quite a bit of damage because they are insecure to man in the middle attacks and even network sniffing in some cases. Man in the middle and sniffing attacks are useless on an encrypted protocol.	HTTP should be replaced with HTTPS to encrypt traffic to the message board FTP should be replaced with SFTP or FTPS to encrypt and file transfers SMTP should implement SSL and IMAP TELNET should be replaced with SSH as TELNET is plaintext.
9.Cross site scripting 	`<script>alert('xss');</script>` Cross site scripting is a huge security concern even on an internal box because an attacker using a tool such as BEEF can do everything from phishing attacks to session hijacking to distributing malware. Because this attack is a reflective attacks it will affect other boxes on the network that connect to the message board.	The best solution to cross site scripting is to escape or sanitize user entry making it impossible for the server to serve xss infected pages to other users.