# Debian Checklist

*Change Root Password*

*Passwd username*

# *Configure IPTables:*

# *su*

*Run iptables -L to check current list. Do iptables -F to flush if there are rules there.*

# *iptables –I INPUT –p tcp --match multiport --dports 80,443,3306 –j ACCEPT*

*iptables –A INPUT –p udp --dports 1514,514 –j ACCEPT*

*Iptables -A INPUT -m state –state ESTABLISHED, RELATED -j ACCEPT*

*iptables –P INPUT DROP*

*sudo bash –c “iptables-save > /etc/iptables.rules”*

*sudo bash –c “iptables-restore < /etc/iptables.rules”*

*drop users*

*iptables -A INPUT -s 192.168.5.179 -p tcp --dport 22 -j DROP*

*if you want to list by line:*

*sudo iptables –L --line-numbers*

*Delete a line:*

*Sudo iptables –D INPUT (line-number)*

*Backup etc and var*

*cp -a /etc /etc.back*

*cp -a /var /var.back*

*MySQL Secure Installation*

*./usr/bin/mysql\_secure\_installation (run find / -name “mysql\_secure\*” if you can't find it)*

*Change root mysql password*

*Change Mysql User Connections*

*nano var/www/config/config.inc.php*

*find* [*root'passwordhere'@localhost*](about:blank) *and change it to the changed mysql password ^*

MD5Deep

* apt-get install md5deep
* md5deep -re \* > Hashes

Check Sudoers file for Domain users

* /etc/sudoers
* /etc/sudoers.d/<filename>

# *Delete unnecessary users*

*Change Default Editor*:

* export EDITOR=nano

*Check users by:*

Egrep –v “^.+:[!\*]:” /etc/shadow | cut –d : -f 1

*You can see users logged in by using*

* w

*Remove users by:*

* Sudo userdel username

*OR*

*Lock user accounts by:*

* vipw –s (s is for /etc/shadow use p for /etc/passwd)
* Put a ! in the 2nd column to lock an account or \* for a service account

*Add new account for SSH:*

* Adduser lucious
* “Luc!0u$\_ Ly0N!\_”

# *Configure OS Sec:*

*sudo apt-get update*

*#sudo apt-get install curl*

*#curl –O*

*#http://ossec.github.io/downloads.html*

wget https://github.com/ossec/ossec-hids/archive/2.9.0.tar.gz

*tar –zxvf ossec-hids-2.9.0.tar.gz*

*cd ossec-hids-2.9.0*

*bash install.sh*

*Follow instructions*

*(if fails, make sure to install c/gnu compilers)*

*Sudo apt-get install build-essential*

*Sudo apt-get install zlib-devel (or libz-devel or libssl-devel)*

*Adding Agents*

*./bin/manage\_agents*

*A*

*Enter Name of Machine*

*Enter IP Address of Machine*

*Hit enter to keep ID*

*Extracting Keys(Client Side)*

Putty into IP address

Use lucious for login, password is “Luc!0u$\_ Ly0N!\_” (no quotes, include space)

su (super user into root, ask for root password)

./*ossec-hids-2.8.1/bin/manage\_agents*

*E*

*Use ID number for your Machine*

*Copy key into your OS Sec client install.*

*openssl genrsa -out /var/ossec/etc/sslmanager.key 2048  
openssl req -new -x509 -key /var/ossec/etc/sslmanager.key -out /var/ossec/etc/sslmanager.cert -days 365*

/var/ossec/bin/ossec-authd -p 1515 >/dev/null 2>&1 &

*Restart OS Sec once clients have entered keys*

*cd /var/ossec*

*./ossec-control restart*

*Modify SSH Config*

*nano /etc/ssh/sshd\_config*

*make sure settings are:*

*Protocol 2*

*SyslogFacility AUTHPRIV*

*LogLevel VERBOSE*

*MaxAuthTries 3*

*X11Forwarding no*

*HostbasedAuthentication no*

*PermitEmptyPassword no*

*PermitUserEnvironment no*

*PermitRootLogin no*

*Save File*

*Modify Access.Conf File:*

*nano /etc/security/access.conf*

*+ : root : cron crond :0 tty1 tty2 tty3 tty4 tty5 tty6*

*- : ALL : ALL*

*(+ = allow; - = deny)*

*(ADD ANY USERS HERE THAT NEED TO SSH IN BY INJECT ENGINE)*

*Modify Pam.D Config Files:*

*nano /etc/pam.d/login*

*Remove # from “account required pam.access.so”*

*nano /etc/pam.d/sshd*

*Remove # from “account required pam.access.so”*

*Restart SSH Service*

*service ssh restart*

*Allow SSH Connections in Iptables:*

*sudo iptables –A INPUT –p tcp -dport 22 –j ACCEPT*

*check /etc/network/if.up.d*

*/etc/network*

*Disable Unnecessary Services*

*List all running processes*

*ps -ax | ps -aux*

*netstat -anp | netstat -tulpn*

*Stop running services that are unnecessary*

*service stop “name of service”*

*service start “name of service” – IF NEEDED*

*update-rc.d openbsd-inet.d disable*

# *Change Directory permissions to root:*

# *chown root:root /etc/fstab*

# *chmod 644 /etc/fstab*

*chown root:root /etc/group*

*chmod 644 /etc/group*

*chown root:root /etc/shadow*

*chmod 400 /etc/shadow*

*chown root:root /etc/apache2*

*chmod 750 /etc/apache2*

*Check permissions on existing files:*

*find / -perm -0002 -type f (file type)*

*find / -perm -0002 -type d (directory type)*

*find / -perm -4000 -type f (SUID)*

*find / -name “ “ 2> /dev/null (Send errors to hell)*

*app / directory / permissions /*

Tripwire

* Install Tripwire
  + apt-get install tripwire
  + use passphrases and site keys
    - do or do not there is no try
    - these are not the droids you are looking for
* Initialize Database
  + tripwire --init
* Run tripwire integrity check
  + tripwire --check
* Print Reports
  + twprint -m r --twrfile /var/lib/tripwire/report/<name>.twr | more
* Update database after an integrity check
  + tripwire --update --twrfile /var/lib/tripwire/report/<name.twr
* You can run tripwire checks from cron in /etc/cron.hourly/tripwire-check.sh with the following:
  + #!/bin/sh
  + HOST\_NAME=`uname -n`
  + if [ ! -e /var/lib/tripwire/${HOST\_NAME}.twd ]; then
  + echo "Error: Tripwire database for ${HOST\_NAME} not found"
  + echo "Run "/etc/tripwire/twinstall.sh" and/or "tripwire --init""
  + else
  + test -f /etc/tripwire/tw.cfg && /usr/sbin/tripwire --check
  + fi

[*https://www.debian.org/doc/manuals/securing-debian-howto/ch4.en.htmlpar*](https://www.debian.org/doc/manuals/securing-debian-howto/ch4.en.htmlpar)

[*http://www.ducea.com/2007/12/05/howto-disable-a-user-account-in-linux/*](http://www.ducea.com/2007/12/05/howto-disable-a-user-account-in-linux/)

*nmap -sT -sV -v -oA mylogs 172.21.0.0/24*

*ipcalc to calculate subnet masks*

*Nick’s notes*

Firewall rules need to be consistently checked on Linux. Make sure there aren't any lingering firewall rules left by the VM creators

* + View rules: iptables -nvL or iptables -L
  + I recommend creating a firewall file that can be used, and then the command: iptables-restore < /location/to/file
* Make sure someone is actively monitoring OSSEC alerts.log
* Make sure the proper library packages are installed before installing OSSEC. On Ubutnu or CentOS packages such as:
  + build-essential
  + zlib-devel or libz-devel
  + libssl-devel (or openssl-devel)
* Nmap/Zenmap scans should be run frequently as you guys are hardening systems
  + Your first run scans should be SYN or TCP connect scans and should do version detection. The T4 option makes the scan more aggressive.
    - nmap -sS -sV -v -O -T4 <your subnet>
    - nmap -sT -sV -v -O -T4 <your subnet>
  + You should save off this first scan data to a file so you can refer to it
  + Scans thereafter should be fast and simply verifying if ports have been closed correctly, or ports have been open
    - nmap -sS -v -T4 <your subnet>
    - nmap -sT -v -T4 <your subnet>
  + Run these scans frequently! During the competition, it was discovered that a telnet port had been left open for about 1.5 hours during the competition
* Make sure you verify the users installed/enabled on every Linux system. Disable OBVIOUS suspicious accounts. Don't disable accounts such as 'daemon', 'sys', or 'syslog'
  + This should be checked by viewing /etc/passwd
  + Lock accounts: passwd -l <user account>
  + You can also disable the account by putting a '!' mark in the password column of /etc/shadow for the user. Make sure you edit /etc/shadow by typing the command 'vipw -s'. Dont edit the file directly. Kwan can talk more about this
    - scsuser:!:16370:0:99999:7:::
  + When you're running commands like 'vipw -s' it's going to choose a default editor, such as vim, which you may not be familiar with. You can change this by setting a temporary environment variable:
    - export EDITOR=nano
* Use your 'ps' command a lot and make sure they're aren't suspicious programs running. For example, during the competition, they had symlinked a program to netcat which was opening an completely unrestricted port. You can run a command like the following:
  + ps auxwf |less -+S
  + The 'less -+S' part helps with word-wrapping
* Your BIGGEST security issues were actually the web applications. Make sure you focus on Joomla, WordPress, PHPMyAdmin, Webmin early in the game.
  + Become familiar with the Apache config files. You can temporarily disable these applications via Apache while you're securing the system. They should not be left open. Common locations for this file could be
    - /etc/apache2/site-enabled/\* (this directory may have numerous configs that need to be reviewed
* Add 'PermitRootLogin no' to your SSHD config file! It wasn't in the checklist (I'll send an example config file)
* Make sure you are viewing running cron jobs. I cannot emphasize this point enough. You need to make sure their are not any scheduled cron jobs (same for scheduled tasks on Windows)
  + crontab -l for each user
  + Look at all directories under /etc/cron\*
    - /etc/cron.d
    - /etc/cron.weekly
    - /etc/cron.hourly
    - /etc/cron.monthly
    - /etc/cron.daily
  + Cron jobs can be disabled either multiple ways:
    - As a specific user: crontab -e
      * This opens the users crontab. Simply comment out the cron job
    - In each of the directories mentioned above, you can disable the cron job script by running:
      * chmod -x /etc/cron.weekly/script
      * This disables cron from executing the job
* Make sure everyone is reviewing log files!
  + Files under /var/log
  + OSSEC will monitor most of these but its a good idea.

Continuously monitor netstat output to check for things like netcat running

* + netstat -tulpc
  + This will show you processes running and what ports are associated. It runs this command every second
* You guys aren't using netcat. I would recommend breaking it on your systems by finding every instance of netcat and preventing execution. Run the commands
  + which netcat
  + which nc
  + which ncat
    - For each one of these
      * chmod a-x <file>
  + You could also remove the package
    - dpkg -l |grep netcat
    - apt-get remove --purge <package>
    - yum remove <package>

SAMPLE SSHD\_CONFIG

-----------------------------------------------

AddressFamily inet  
Protocol 2  
  
SyslogFacility AUTHPRIV  
LogLevel VERBOSE  
  
X11Forwarding no  
MaxAuthTries 3  
IgnoreRhosts yes  
HostbasedAuthentication no  
PermitRootLogin no  
PermitEmptyPasswords no  
PermitUserEnvironment no  
Ciphers aes128-ctr,aes192-ctr,aes256-ctr  
Port 22  
HostKey /etc/ssh/ssh\_host\_rsa\_key  
HostKey /etc/ssh/ssh\_host\_dsa\_key  
HostKey /etc/ssh/ssh\_host\_ecdsa\_key  
UsePrivilegeSeparation yes  
KeyRegenerationInterval 3600  
ServerKeyBits 768  
LoginGraceTime 120  
StrictModes yes  
RSAAuthentication yes  
PubkeyAuthentication yes  
RhostsRSAAuthentication no  
ChallengeResponseAuthentication no  
X11DisplayOffset 10  
PrintMotd no  
PrintLastLog yes  
TCPKeepAlive yes  
AcceptEnv LANG LC\_\*  
Subsystem sftp /usr/lib/openssh/sftp-server  
UsePAM yes