UMSL CCDC Blue Team Manual

v1.0.02052024

Authored By: Chris Suhre

~ Table of Contents ~

- System Updates/User & File Permissions 2-3
- Firewall Configurations 3-4
- System Monitoring 4-6
- Network Monitoring 7-9
- Services 9-17
- File Editors 17-19

==========System Updates/User & File Permissions==========

For Ubuntu:
sudo apt-get update
Remove Packages no longer needed:
sudo apt-get autoremove

For CentOS/Fedora:
sudo yum update
…Some CentOS systems may require this:
sudo yum install epel-release

#Change to root user:
su

Change root password:
sudo passwd root

Change user password (Ex: user = john):
sudo passwd john

Change or lock user account:
sudo passwd -l <username>

Delete user account:
sudo userdel <username>

Add group:
groupadd <group name>

Add user:
useradd <user name>

Modify user's account detail
sudo usermod <option> USERNAME
Ex: usermod -a - G dev joe
…This command instructs the system to add joe to the dev group

Create directory:
mkdir -p /data/dev
…mkdir = command used to create directories
…-p = option tells mkdir to make parent directories
…data/dev = specifies the path of the new directory/directories you want to create, it's creating dev directory inside /data

Change group ownership:
chown :groupname filename
Ex: chown :dev dev
chown = command used to change owner and/or group of file or directory
:dev = group ownership
dev = target of the chown command, either a filename or directory
#Change group ownership with specified username:
chown username:groupname filename
Change file permissions:
chmod <u/g/o> <+/-> <r/w/x>
…chmod = command to change file permissions
…<u/g/o> = u → user, g → group, o → others
…<+/-> = + → add, - → remove
…<r/w/x> = r → read, w → write, x → execute
Ex: chmod g+w *
…add write permissions for the group, * = match all files and directories in the current directory

GitHub repository clone:
git clone https://github.com/example/repo.git
…the URL must end in .git
...if git is not downloaded on system use:
sudo apt install git

Execute a Bash script:
bash <scriptname.sh>
…can execute the script in the bash terminal if you navigate to the directory the script is located
…if script is not executable use:
chmod +x <scriptname.sh>
Then run the Bash execute command

#Purge insecure services (Ubuntu if needed):
sudo apt-get --purge remove xinetd nis yp-tools tftpd atftpd tftp d-hpa telnetd rsh-server rsh-redone-server

==========Firewall Configurations==========

List file path iptables is listed under:
which iptables

Install iptables (if needed):
sudo apt-get install iptables

List all firewall rules:
sudo iptables -L -v

Display Firewall rules with line numbers:
sudo iptables -L --line-numbers

To flush the entire iptables and start fresh:
sudo iptables -F

iptables syntax using Append (-A):
sudo iptables -A <chain> -p protocol (tcp/upd) --dport <port no.> -j <target>
Ex (443 = https):
sudo iptables -A INPUT -p tcp --dport 443 -j ACCEPT
…order matters in a firewall table, system will read the table from top to bottom
…chains explained:
INPUT = configure any connection coming into the system
FORWARD = if system is just a forwarder, configure this chain
OUTPUT = block any traffic going out of system, configure this chain
-A = append the entry into the next available line in the table → this is why order is important

iptables syntax using Insert (-I) line:
sudo iptables -I INPUT 1 -p tcp --dport 80 -j ACCEPT
…-I = insert option
…INPUT 1 = Chain + Line Number to specify which line of table to insert this rule

iptables Drop line:
sudo iptables -A INPUT -j DROP
…this will drop all traffic for protocols not specified in the table before this line
…IMPORTANT: placing drop in the last line of the firewall table is critical!

Delete a specific line (Ex: Line 3):
sudo iptables -D INPUT 3

Drop traffic to a specific IP address:
sudo iptables -A INPUT -s <IP Address> -j DROP
…drop traffic coming into the system from the specified IP address

Enable iptables on reboot:
sudo systemctl enable iptables

==========System Monitoring==========

What processes are running?
ps -aux
OR
ps -ea

Search for ps running as a specific user (Ex: root):
ps -U root -u root u

To search for a specific service (Ex: apache2):
ps -ea | grep apache2

Check status of a service (Ex: ssh):
systemctl status ssh

Remove a service (Ex: fail2ban):
sudo apt-get remove fail2ban

***** Auth.log *****
# Display entire auth.log file:
sudo cat /var/log/auth.log

Tail the auth.log file:
sudo tail -f /var/log/auth.log

Search for specific entries (Ex: sshd):
sudo grep 'sshd' /var/log/auth.log
...Replace 'sshd' with specific name for search

Search for failed login attempts:
sudo grep 'Failed' /var/log/auth.log

Check for sudo command usage:
sudo grep 'sudo' /var/log/auth.log

Display entries for specific user (Ex: user = john):
sudo grep' john' /var/log/auth.log

Show recent successful logins:
sudo grep 'Accepted' /var/log/auth.log

Check for new user or group creation:
sudo grep 'new user' /var/log/auth.log
...search for new user
sudo grep 'new group' /var/log/auth.log
...search for new group

***** Syslog *****
# View most recent log messages:
sudo tail -f /var/log/syslog

Display specific log entries (Ex: ssh):
sudo grep ssh /var/log/syslog

Restart syslog service:
sudo systemctl restart rsyslog

Alternative navigation to syslog and read:
cd /var/log
ls
cat syslog | less
...press q to exit view
cat syslog | more
...#press q to exit view
cat syslog | tail
...#press q to exit view
cat syslog | head
...#press q to exit view

*****lsof*****
List Open Files (lsof)
sudo lsof

Identify Open Files by a User (Ex: user = john):
sudo lsof -u john

Find Files Opened By a Process ID (Ex: PID = 5036:
sudo lsof -p 5036

List Network Connections:
sudo lsof -i

Monitor Files in a Directory (Ex: directory = /var/log:
sudo lsof +D /var/log

Find Open Files on a Specific Port (Ex: portnumber = 443):
sudo lsof -i 443

List IPv4 and IPv6 Network Files:
sudo lsof -i 4
sudo lsof -i 6

Show Processes Listening on Ports:
sudo lsof -i -sTCP:LISTEN

Display Files Opened by a Service (Ex: service = sshd):
sudo lsof -c sshd

Find Deleted Files Still in Use:
sudo lsof | grep deleted

==========Network Monitoring==========

Install net-tools (if not already on system):
sudo apt-get install net-tools

Display Network Connections:
netstat -tuplan
...block any unknown ports immediately using firewall
...locate daemon in /etc folder and delete the whole system
...can tell service to stop

Terminate Process ID (Ex: PID = 4828)
sudo kill -9 4828

*****Tool: NMAP*****
sudo nmap <IP Address>
...lists ports, state, and services running on the specified IP address
...nmap will only scan 1,000 most commonly used ports, not the full 65K available (we can do that but not necessary in most cases)

Can scan a range of IP addresses if desired, must know network subnet mask:
sudo nmap <IP Address/subnet mask>
Ex: sudo nmap 192.168.1.40/24
...lists each host with their ports, state, and services running

Different NMAP scan types (get same results just different scans, ex: TCP or UDP specific scans)
sudo nmap -sS <IP Address>
-sS = TCP SYN scan (most popular, scans quickly for ports not protected by firewall; only performs step 1 of TCP 3 way handshake)

sudo nmap -sT <IP Address>
-sT = establishes full TCP 3 way handshake, no sudo needed
--> will leave a trace that is easily detected, hence -sS is better option

sudo nmap -sU <IP Address>
-sU = UDP scan; Unpopular as most services use TCP; However, could be exploited if services are using UDP; Takes longer to complete
...pressing up arrow key will show % of scan completed so far

NMAP Manual:
man nmap
...detail about each option for nmap

nmap -sA
-sA = SYN/ACK
...can be useful to map out a firewall

NMAP - Discovering Target OS:
sudo nmap -O <IP Address>
... -O = OS testing
...OS must have at least one port open and closed
#Output Explained
...Will get list of open ports; MAC Address with VM noted - useful for hackers to detect honeypots which use similar setups
...Type of OS with version; Network Distance: 1 hop = 1 hop indicates it is within your network

NMAP - Detecting Version of Service Running on an Open Port
sudo nmap -sV <IP Address>
...This will give you Port, State, Service, and Version
...Can simply copy paste version of a service into google to find exploitables

sudo nmap -A <IP Address>
…Output Explained:
...-A = enables NMAP script scanning; Also enables detecting OS and service versions with much greater detail
...Output of different scripts running on target
...The most output and easily detectable if target system has security measures

NMAP - Version Intensity:
sudo nmap -sV --version-intensity 9 <IP Address>
...Default --version-intensity = 7; Increasing intensity will take longer; Not necessary when scanning from local network

Other Useful NMAP options:
sudo nmap -sn <IP Address>
...-sn option = perform same function as Netdiscover tool to see what hosts are up

sudo nmap -p 80 <IP Address>
...-p <port #,#s, or range>
…- p option = what range of ports do you want to scan with NMAP
…Use port scan of only http port 80

sudo nmap -p 80,22,100 <IP Address>
…Multiple port scans at the specific IP address

sudo nmap -p 1-100 <IP Address>
…Port scan range (you can specify all 65K ports to scan if needed)

sudo nmap -F <IP Address>
...-F option = scan first 100 most used ports

NMAP - Send output to a file (Ex: file name = outputofscan.txt):
sudo nmap -sS <IP Address> >> outputofscan.txt

#NMAP - Send output to a file and terminal:
sudo nmap -oN output <IP Address>

==========Services==========

***** Apache2/HTTPD/NGINX*****

Install apache2/httpd:
sudo apt install apache2
sudo yum install httpd

Start apache/httpd service:
sudo systemctl start apache2
sudo systemctl start httpd

Enable apache/httpd service on boot:
sudo systemctl enable apache2
sudo systemctl enable httpd

Check apache/httpd status:
sudo systemctl status apache2
sudo systemctl status httpd

Restart apache/httpd:
sudo systemctl restart apache2
sudo systemctl restart httpd

Update firewall rules to enable http and https (if needed)*

Path to apache2 config file: /etc/apache2
Path to apache 2 html file: /var/www/html
…# make a backup of the index.html file…
Path to httpd config file: /etc/httpd

Make a backup of var/www directory:
cd /var/www
cp -r html /home/sysadmin/
…# replace /sysadmin with name of user

Remove original directory:
sudo rm -r /path/to/original_directory
…# Ex: sudo rm -r /var/www/html
[CONT.]
Move backup directory:
sudo mv /path/to/backup /var/www
…Ex: sudo mv /home/sysadmin/html /var/www

Disable trace for apache2:
cd /etc/apache2/conf-available
sudo gedit security.conf
…# Update…
Uncomment TraceEnabled Off
Comment TraceEnabled On
#ServerTokens OS —> uncomment & change to = ServerTokens Prod
…# Save…

Update apache2 config:
sudo gedit apache2.conf
…Edit <Directory /var/www/> line (add line if not there)
Options -Indexes -FollowSymLinks
…# Reload apache2 configs #
…Install mod security
cd mods-enabled
…then…
sudo apt install libapache2-mod-security2 -y
…enable security2…
sudo a2enmod security2
…restart…
sudo systemctl restart apache2
…Install evasive security mod…
sudo apt install libapache2-mod-evasive -y
…enable evasive…
sudo a2enmod evasive
…restart…
sudo systemctl restart apache2

Reload apache2/httpd after config updates:
sudo systemctl reload apache2
sudo systemctl reload httpd

Install nginx (if needed):
sudo apt install nginx
sudo yum install nginx

Path to nginx config file: /etc/nginx/nginx.conf
..# Add or modify the following line in the http block of config file…
server_tokens off;
…# Limit size requests in http or server block (if needed)...
client_max_body_size 10M;

Start and enable nginx services:
sudo systemctl start nginx
sudo systemctl enable nginx

Update firewall rules to enable http and https (if needed)*

PHP security hardening
…# Edit this file php.ini
…# File path: /etc/php
cd /etc/php
sudo gedit php.ini
…# Disable these functions…
disable_functions = exec, system, shell_exec, passthru, phpinfo, show_source, popen, proc_open
…# Limit file uploads
file_uploads = Off
…# Set appropriate permissions
open_basedir = "/var/www/html"
…# Error reporting
display_errors = Off
log_errors = On
error_log = /var/log/php/error.log
…# Session security
session.cookie_httponly = 1
session.cookie_secure = 1
session.use_only_cookies = 1
session.use_strict_mode = 1

POP3 Server (Dovecot) config file path: /etc/dovecot/conf.d/10-auth.conf
…# Make a copy of the config file!
…# Edit config file
…# Disable plaintext authentication
disable_plaintext_auth = yes
…# Restart Dovecot to apply changes
sudo systemctl restart dovecot

***** Curl *****
# Install curl:
sudo apt-get install curl

Check website's HTTP headers:
curl -I http://example.com
...#insert applicable url

Test SSL/TLS certificates:
curl --insecure -v https://example.com 2>&1 | grep -i "expire date"
...#insert applicable url

Using curl to TRACE:
curl -v -X TRACE https://www.example.com
...#insert applicable url

***** Clamav *****
# Install clamav:
sudo apt-get install clamav
sudo yum install clamav

Stop clamav:
sudo systemctl stop clamav-freshclam

Update clamav database:
sudo freshclam

Start service with updated database:
sudo systemctl start clamav-freshclam

Keep clamav running after reboots:
sudo systemctl enable clamav-freshclam

Running a clamav scan:
sudo clamscan -r /
Backup original clamavd.conf:
cd /etc/clamav/
...#change to the right directory
sudo cp clamd.conf clamd.conf.backup

Update/Review clamav configs:
sudo gedit /etc/clamav/clamd.conf
...Update these lines (if needed)...
ScanPe true
ScanELF true
DetectPUA true
ScanMail true
HeuristicScanPrecedence true
PhishingScanURLs true

Restart clamav:
sudo systemctl restart clamav-daemon

View clamav logs:
sudo cat /var/log/clamav/clamav.log
If a virus is detected, note the path and names of the file(s)
Remove infected file:
sudo rm /path/to/infected/file
...after removal, run a follow-up scan

***** Auth.log & Access.log (Apache2)*****
# Display entire log file:
sudo cat /var/log/auth.log
sudo cat /var/log/apache2/access.log

Tail the auth.log file:
sudo tail -f /var/log/auth.log

Search for specific entries:
sudo grep 'sshd' /var/log/auth.log
...#Replace 'sshd' with specific name for search

Search for failed login attempts:
sudo grep 'Failed' /var/log/auth.log

Check for sudo command usage:
sudo grep 'sudo' /var/log/auth.log

Display entries for specific user:
sudo grep 'username' /var/log/auth.log
...#enter the specific username

Show recent successful logins:
sudo grep 'Accepted' /var/log/auth.log

Check for new user or group creation:
sudo grep 'new user' /var/log/auth.log
...#search for new user:
sudo grep 'new group' /var/log/auth.log
...#search for new group

***** Fail2ban *****
# Install Fail2Ban:
sudo apt-get install fail2ban

Config File location: /etc/fail2ban/jail.conf

Make copy of config file and rename to local to override settings in jail.conf:
sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local
...fail2ban will use local file over original jail.conf file

Edit Fail2ban local jail config file:
sudo gedit jail.local
...# Add ip to ignore ip in jail.local file...
ignoreip = 127.0.0.1/8 <insert IP addr>
...# Ex: ignoreip = 127.0.0.1/8 192.168.116.130/24
...# add ip to line and separate each with a space
...# Check ip addr on system:
ip addr

Set Ban Time and Retry Limits:
[Default]
bantime = 3600 #1 hour (duration in seconds for IP banning)
findtime = 600 #10 minutes (duration in seconds during which multiple failed logins trigger a ban)
maxretry = 5 (number of failures before an IP is banned)

Monitor Specific Services:
[service name] #header for jail config
enabled = true
...#set to true you want fail2ban to monitor this specific service
port = ssh
filter = sshd
logpath = /var/log/auth.log
maxretry = 3

Restart Fail2ban service:
sudo service fail2ban restart

Check Fail2ban service status:
sudo systemctl status fail2ban

List active jails:
sudo fail2ban-client status

List banned IPs for a specific jail:
sudo fail2ban-client status <service name>
...# Ex: sudo fail2ban-client status sshd

List iptables rules where banned IPs are visible:
sudo iptables -L -n
...# Look for rules under 'Chain fail2ban-<jailname>

Fail2ban log files location: /var/log/fail2ban.log
Review auth files in this location: /var/log/auth.log

***** Check for Rootkits *****
# Install rootkit checker rkhunter:
sudo apt-get install rkhunter
...select Ok and No configuration in the menu options

Update rkhunter's data files:
sudo rkhunter --update

Run manual rkhunter scan:
sudo rkhunter -C

Run manual rkhunter scan with more detail:
sudo rkhunter --check

rkhunter properties update:
sudo rkhunter --propupd

rkhunter list rootkits being checked for:
sudo rkhunter --list rookits

View rkhunter scan reports:
sudo cat /var/log/rkhunter.log

***** Auditd Tool *****
# Install auditd:
sudo apt-get install auditd

Start auditd:
sudo systemctl start auditd

Enable auditd to start on reboot:
sudo systemctl enable auditd

Check auditd status:
sudo systemctl status auditd

...# Monitor access to important files...
Audit for any changes to password file:
sudo auditctl -w /etc/passwd -p wa -k password-file

#Audit to monitor user and group changes:
sudo auditctl -w /etc/group -p wa -k group-file
sudo auditctl -w /etc/gshadow -p wa -k shadow-file

Audit to monitor sudoers file:
sudo auditctl -w /etc/sudoers -p wa -k sudoers-file

Audit to monitor network configuration changes:
sudo auditctl -w /etc/sysconfig/network -p wa -k network-change

Reload auditd configs after changing rules:
sudo systemctl restart auditd

Review audit logs:
sudo sudo ausearch -k rule_key
...#replace rule-key with specified rules
...#Ex: sudo ausearch -k sudoers-file

#Search from audit control:
sudo ausearch -f /etc/passwd

***** OPENSSH *****
# Install opeenssh-server:
sudo apt install openssh-server
sudo yum install openssh-server

Update ssh config files:
cd /etc/ssh
...#navigate to the ssh directory
Two config files:
#1. sssh_config
#2. sshd_config
...# Run this command to edit the sshd config...
sudo gedit sshd_config &
...# Update to remove root login...
PermitRootLogin no
...# Close window to save changes...
Restart ssh service:
sudo service ssh restart
Check status of ssh:
systemctl status ssh

***** Awk *****
#Check shadow file where passwords need to be flagged if they have root privileges:
awk -F: '($2 == "0") {print}' /etc/shadow

#Check password file for flagging users with root privileges:
awk -F: '($3 == "0") {print}' /etc/passwd

sudoers
File path: /etc/sudoers

#Command to edit:
sudo visudo

==========File Editors==========

Vi Commands
#Basic Movement
h: Move left
j: Move down
k: Move up
l: Move right
0: Move to the beginning of the line
^: Move to the first non-blank character of the line
$: Move to the end of the line
G: Move to the last line of the file
gg: Move to the first line of the file
w: Move forward to the start of the next word
b: Move backward to the start of the previous word

#Insert Mode
i: Enter insert mode before the cursor
I: Enter insert mode at the beginning of the line
a: Enter insert mode after the cursor
A: Enter insert mode at the end of the line
o: Open a new line below the current line and enter insert mode
O: Open a new line above the current line and enter insert mode

#Editing
x: Delete the character under the cursor
dd: Delete the current line
dw: Delete from the cursor to the end of the word
d$: Delete from the cursor to the end of the line
u: Undo the last operation
Ctrl + r: Redo the last undo

#Copying and Pasting
yy: Yank (copy) the current line
yw: Yank (copy) from the cursor to the end of the word
p: Paste the yanked text after the cursor
P: Paste the yanked text before the cursor

#Searching and Replacing
/pattern: Search for pattern
?pattern: Search backward for pattern
n: Repeat the last search in the same direction
N: Repeat the last search in the opposite direction
:%s/old/new/g: Replace all occurrences of old with new in the file

#File Operations
:w: Save the file
:q: Quit Vi/Vim
:wq or :x: Save and quit
:q!: Quit without saving

#Miscellaneous
Ctrl + g: Show current file name and status
.: Repeat the last command
:%!fmt: Format the entire file
:set number: Show line numbers
:set nonumber: Hide line numbers

Nano Commands
Overview of nano's shortcuts
The editor's keystrokes and their functions

File handling
Ctrl+S Save current file
Ctrl+O Offer to write file ("Save as")
Ctrl+R Insert a file into current one
Ctrl+X Close buffer, exit from nano

Editing
Ctrl+K Cut current line into cutbuffer
Alt+6 Copy current line into cutbuffer
Ctrl+U Paste contents of cutbuffer
Alt+T Cut until end of buffer
Ctrl+] Complete current word
Alt+3 Comment/uncomment line/region
Alt+U Undo last action
Alt+E Redo last undone action

Search and replace
Ctrl+Q Start backward search
Ctrl+W Start forward search
Alt+Q Find next occurrence backward
Alt+W Find next occurrence forward
Alt+R Start a replacing session

Deletion
Ctrl+H Delete character before cursor
Ctrl+D Delete character under cursor
Alt+Bsp Delete word to the left
Ctrl+Del Delete word to the right
Alt+Del Delete current line