****

# **Wildcard CCDC Linux Checklist**

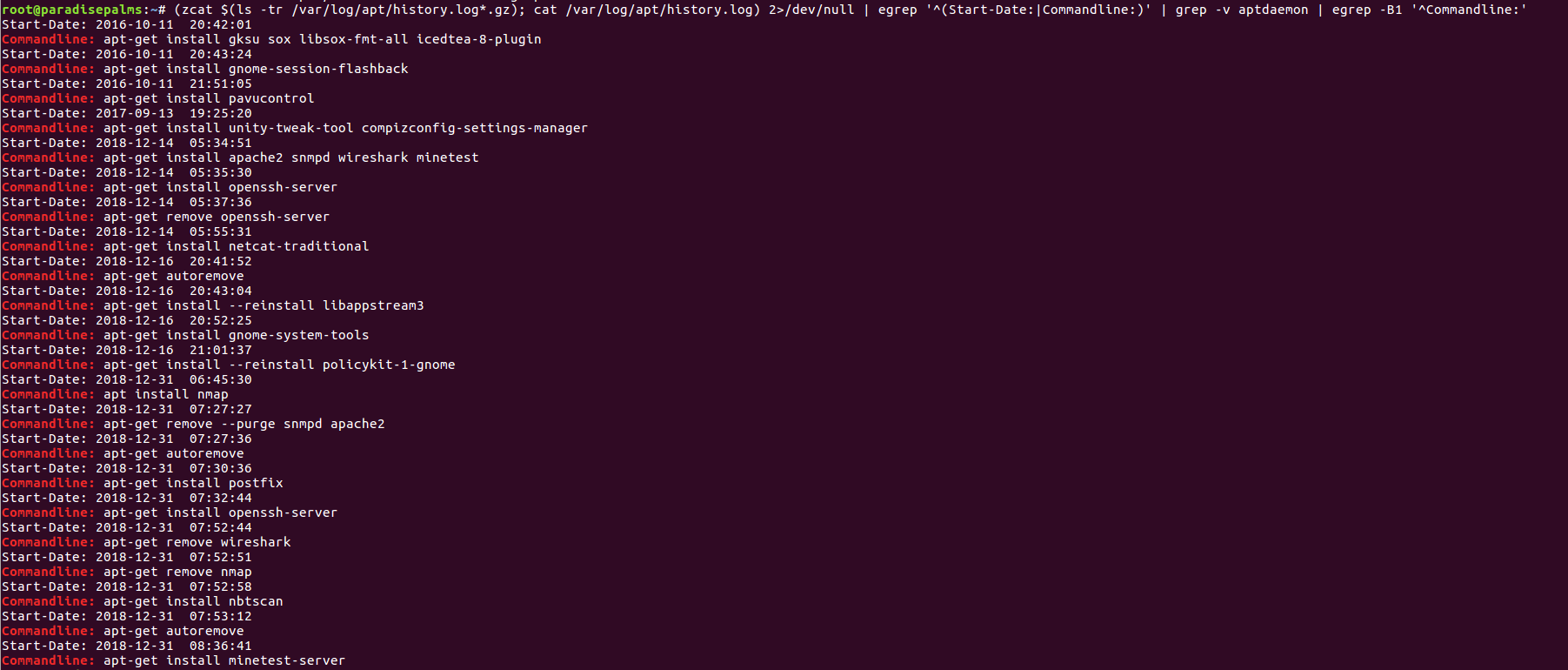
|  |
| --- |
| Key  * Anything in text like this is a command to be typed in terminal   + ping 8.8.8.8 * Anything in single quotes is a file or directory path   + ‘/etc/shadow’ * Anything in double quotes is a line to be added to a file   + “PASS\_MAX\_DAYS 30” |

**Sections**

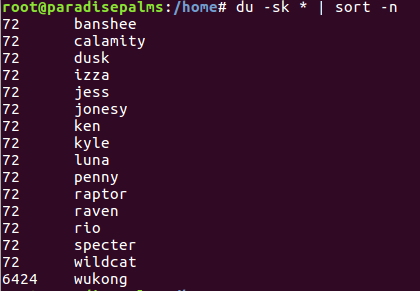
* Pre Image Scanning
* Operating System Updates
* User Auditing
* Password Policies
* Local Policies
* Uncategorized Operating System Settings
* Defensive Countermeasures
* Application Updates
* Prohibited Files
* Prohibited Software
* Malware
* Application Security Settings

**Pre Image Scanning**

* Store any backups and logs in ‘/ccdc’
* If either the authentication log or apt history log outputs anything, save that output in the ‘/ccdc’ folder
* Check all ‘/var/logs/auth.log\*’
  + (zcat $(ls -tr /var/log/auth.log\*.gz); cat /var/log/auth.log; cat /var/log/auth.log.1) 2>/dev/null | grep -i COMMAND
* Check the Apt log at ‘/var/log/apt/history.log’
  + (zcat $(ls -tr /var/log/apt/history.log\*.gz); cat /var/log/apt/history.log) 2>/dev/null | egrep '^(Start-Date:|Commandline:)' | grep -v aptdaemon | egrep -B1 '^Commandline:'
  + Would look like this:



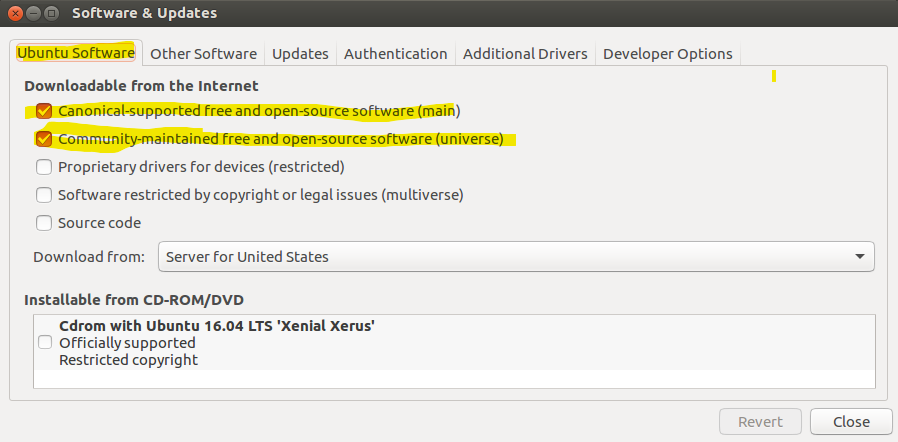
* Check for vulnerabilities with lynis
  + sudo apt-get install -y lynis
    - sudo lynis audit system
* Use linPEAS to check for possible privilege escalation
  + curl -L https://github.com/carlospolop/PEASS-ng/releases/latest/download/linpeas.sh | sh
* Check /home for different folder sizes
  + du -sk \* | sort -n
  + Any directory that isnt the “default” size other than the main user (wukong) has extra files inside, check it out



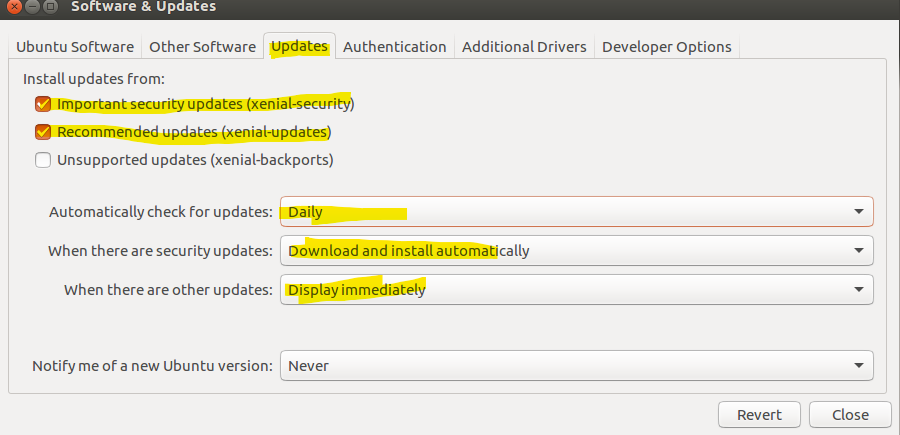
* + - cd /home/{user}
    - tree -a
* Baselines
  + cd /usr/bin
  + ls -al | awk '{print $1, $2, $3, $4, $9}' > /ccdc/bin.current
  + cd /usr/sbin
  + ls -al | awk '{print $1, $2, $3, $4, $9}' > /ccdc/sbin.current
  + cd /etc
  + find /etc -type f -print0 | sort -z | xargs -r0 sha256sum >> /ccdc/etc-hashes.current
  + find /etc -type f -print | sort -z > /ccdc/etc.current
  + apt install debsums
  + debsums -a | grep -v OK > /ccdc/debsums
* Compare your baselines:
  + Download your OS’s versions to /ccdc
  + Compare the hashes of /etc now
    - sha256sum -c /ccdc/etc-hashes.list | grep -v OK > /ccdc/hashes.diff
  + Later on, you can compare the binaries
    - diff /ccdc/bin.current /ccdc/bin.list | egrep "^<"
    - diff /ccdc/sbin.current /ccdc/sbin.list | egrep "^<"
    - diff /ccdc/etc.current /ccdc/etc.list | egrep "^<"

**Operating System Updates**

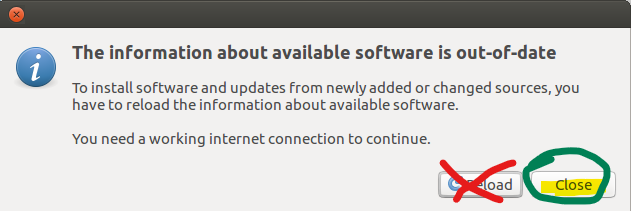
* Run the following command to open the Gnome apt manager
  + software-properties-gtk
  + if it doesn't exist, install it
  + Configure Main and Contrib files



* + Configure security and automatic updates



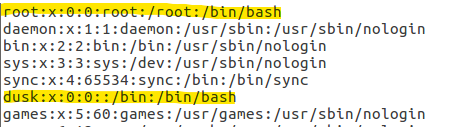
* When closing, do NOT update the package lists



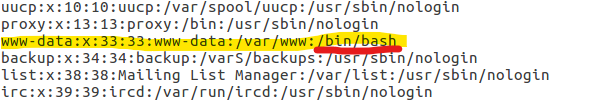
* Install apt fast and update the system
  + apt update
  + apt install curl wget apt-transport-https dirmngr -y
  + /bin/bash -c "$(curl -sL <https://git.io/vokNn>)"
  + apt-fast update -y
  + apt-fast upgrade -y
* Start this first to have it done quickly
* If apt-fast doesnt install, just do apt upgrade

**User Auditing**

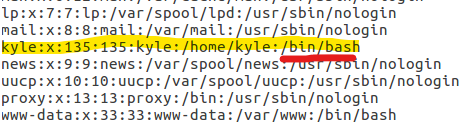
* Check ‘/etc/passwd’ for the following:
  + unauthorized users
    - Any user with a uid above 1000 and not in the readme and not a service user like mysql or ftp
  + root users
    - Any non-root user with UID or GID 0 (ex dusk below)



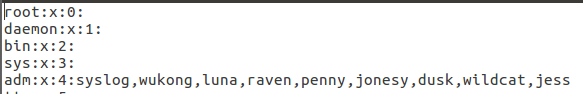
* + system users with login shells or hidden users
    - Any user with uid below 1000 with any sort of login shell like the following
      * /bin/bash
      * /bin/sh
      * /bin/zsh
      * /bin/dash
    - if the user is a default system account like bin or man, change the shell



* + - if the user is not a default system account and is unauthorized, change the uid and delete the user



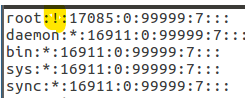
* + password hashes
    - every account should have an “x” between the first and second :
* Check /etc/group for the following:
  + unauthorized admins(sudo adm lpadmin wheel)
    - Check any administrative group for non admin users



* + nopasswdlogin
    - No user should be in this group



* + users in root group
    - no user should be in the root group
* Check ‘/etc/shadow’ for the following:
  + System users with password hashes, should just have \*
  + Root not locked out (should have root:!:)
    - use sudo passwd -l root



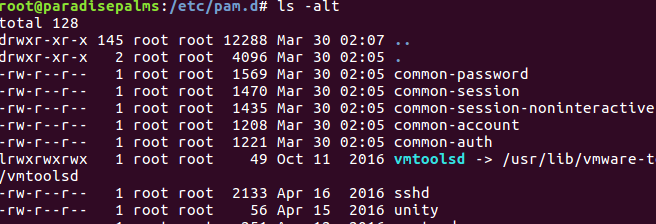
* If you have any issues with managing users, check for immutable or append only files and folders
  + chmod -i /etc/passwd
* DO PASSWORD POLICIES NOW
* Come back afterwards
* To change all authorized users passwords, run the following one liner:
* read; for u in $(cat /etc/passwd | grep -E "/bin/.\*sh" | cut -d":" -f1); do echo "$u:$REPLY" | chpasswd ; done
* When you press enter, it will just be blank, type in the password you want to change it to.
* Check your work with sudo -s in an new terminal

**Password Policies**

* Run sudo apt-get install libpam-pwquality-y
* Open up ‘/etc/login.defs’
  + Find the lines that have “PASS\_MAX\_DAYS”
  + Change to the following:



* + Find the line with “ENCRYPT\_METHOD”
    - Change to “ENCRYPT\_METHOD SHA512”
* Check for modified pam files
  + ls -alt /etc/pam.d



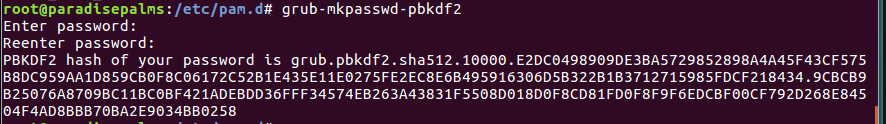
* + Check anything recently edited. For example, ‘gdm-password’ might have “auth required pam\_succeed\_if.so user != root quiet” commented out, if so uncomment it. If you see any lines referencing a specific user, remove them
* Overwrite new pam configurations
  + /etc/pam.d/common-password and common-auth
  + take from the section that matches your OS
  + test this by opening a new shell, and trying to run a sudo command
    - You cannot test if sudoers has a bad entry, so if you arent required your password in the first place, then fix sudoers first

**Local Policies**

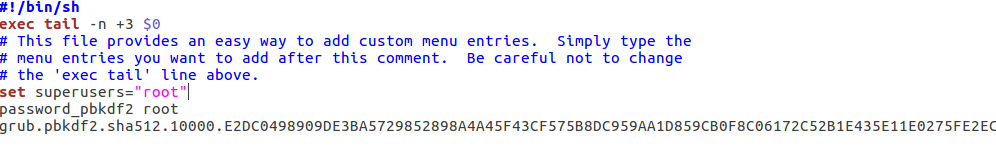
* LightDM (Ubuntu 16 or below, Fedora)
  + Open ‘/etc/lightdm/lightdm.conf’, add the following lines:
    - * [SeatDefaults]
      * user-session=ubuntu
      * greeter-hide-users=true
      * greeter-show-manual-login=true
      * allow-guest=false
      * greeter-allow-guest=false
      * autologin-user=none
      * autologin-guest=false
      * AutomaticLoginEnable=false
      * xserver-allow-tcp=false
    - Check other directories in ‘/etc/lightdm’, delete anything you see that is looks malicious
* GDM3 (Ubuntu 18+, Debian)
  + Open ‘/etc/gdm3/custom.conf’
    - [daemon]:
      * AutomaticLoginEnable=false
      * AutomaticLogin=
      * TimedLoginEnable=false
      * TimedLogin=
      * TimedLoginDelay=10
    - [greeter]
      * IncludeAll=false
      * Exclude=bin,root,daemon,adm,lp,sync,shutdown,halt,mail,news,uucp,operator,nobody,nobody4,noaccess,postgres,pvm,rpm,nfsnobody,pcap
    - [security]
      * DisallowTCP=true
      * AllowRoot=false
      * AllowRemoteRoot=false
      * VerboseAuth=false
    - [xdmcp]
      * Enable=false
    - [chooser]
      * Broadcast=false
    - [debug]
      * Enable=true
* Overwrite ‘/etc/sysctl.conf’ with the following:
  + <https://github.com/klaver/sysctl/blob/master/sysctl.conf>
  + change “kernel.kptr\_restrict=1” to “kernel.kptr\_restrict=2”
  + Add the following lines:
    - net.ipv6.conf.all.disable\_ipv6=1
    - kernel.dmesg\_restrict=1
    - kernel.unprivileged\_userns\_clone=0
  + Run sysctl -p to apply them
  + Also, clear the configurations in /etc/ufw/sysctl.conf

**Uncategorized Operating System Settings**

* Open ‘/etc/sudoers’ by running sudo visudo
  + Make sure no line contains either “!authenticate” or “NOPASSWD”, if a line contains “NOPASSWD:”, then remove the “NOPASSWD:”. If a line contains “!authenticate”, then delete the line.
  + Check all files in ‘/etc/sudoers.d/’ as well
* Set sticky bit on world writable directories
  + df --local -P | awk {'if (NR!=1) print $6'} | xargs -I '{}' find '{}' -xdev -type d -perm -0002 2>/dev/null | xargs chmod a+t
* Open ‘/etc/fstab’ and add the following line to the bottom of it:
  + none /run/shm tmpfs noexec,nosuid,nodev 0 0
  + proc /proc proc rw,nosuid,nodev,noexec,relatime,hidepid=2 0 0
  + /tmp /var/tmp none rw,nodev,noexec,nosuid,bind 0 0
* run mount -a to apply changes
* Grub Bootloader
  + Ensure the file ‘/etc/default/grub’ does not have the line “noexec=off”
  + Secure Grub Bootloader by following these steps:
    - Install/update grub by running apt-get install grub-common -y
    - Run grub-mkpasswd-pbkdf2 and enter in any password (e.g. “CyberPatriotRul3z!”)



* + - The password hash starting with grub.pbkdf will be used below
    - Add the following lines to ‘/etc/grub.d/40\_custom’
      * “set superusers="root"”
      * “password\_pbkdf2 root {password hash from above}”



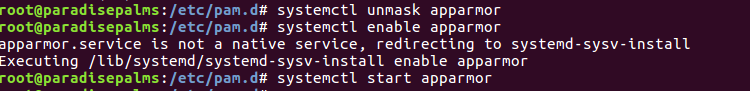
* + - Run update-grub to set these settings
  + Change /etc/grub/defaults
    - GRUB\_CMDLINE\_LINUX="audit=1 audit\_backlog\_limit=8192 apparmor=1 security=apparmor"
* /etc/systemd/resolved.conf
  + add the line “LLMNR=no”
* Delete the file /etc/securetty
* Critical Services
  + Each critical service has configurations and content files that may have insecure permissions
  + Configuration files will be in /etc/servicename
  + Content files:
  + SSH:
    - /root/.ssh/\* /home/\*/.ssh/\*
    - https://www.tecmint.com/set-ssh-directory-permissions-in-linux
  + FTP/SMB:
    - Where the file share is, usually in /srv or /var
    - folders should be 755 and files should be 644
    - Also check for permissions INSIDE of the configuration
  + SQL:
    - /var/lib/mysql/
    - this and any databases below should not be world writable
  + Apache/Nginx
    - /var/www/html

**Defensive Countermeasures**

* Configure UFW
  + sudo apt-fast install ufw gufw
  + sudo ufw enable
  + sudo ufw logging high
* Configure Critical Services (use ports)
  + sudo ufw allow PORTNUM

## **Service Auditing**

* Start the sudo rsyslog, systemd-journald, and apparmor services
  + sudo systemctl unmask {service}
  + sudo systemctl enable {service}
  + sudo systemctl start {service}



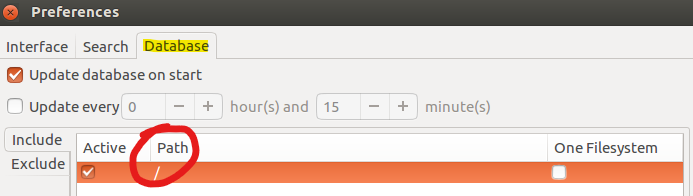
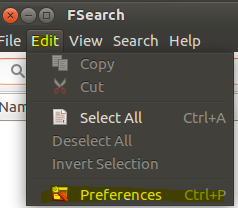
* Check for any unnecessary services
  + sudo service --status-all
  + sudo systemctl list-unit-files | grep enabled
* If any of the following are NOT a critical service, but are installed. remove them
  + Apache2
  + Bind9
  + Dovecot
  + exim4
  + Gerbera
  + Icecast2
  + Inetd
  + IRC daemon
  + Jupyter Notebook
  + MariaDB
  + Minetest
  + Mongodb
  + MySQL
  + NFS
  + Nginx
  + OpenArena
  + opensmtpd
  + Postfix
  + Postgresql
  + ProFTP
  + PureFTP
  + Rsync
  + Samba
  + SNMP
  + Squid proxy
  + Telnet daemon
  + TigerVNC
  + TightVNC
  + VSFTPD
  + WorldForge

**Application Updates**

* Make sure that ALL programs or services mentioned in the readme are updated. DOUBLE AND TRIPLE CHECK
  + apt-mark showhold
  + If there are any packages held, run
    - apt-mark unhold PACKAGE
  + Then rerun apt upgrade

**Prohibited Files**

* Run these commands in terminal
  + sudo add-apt-repository ppa:christian-boxdoerfer/fsearch-stable
  + sudo apt-get update
  + sudo apt install fsearch-trunk
  + fsearch
    - Go to preferences and then add ‘/’ to the database
    - Then search for “.<extension>” in the search bar
      * .mp3
      * .mp4
      * .csv
      * .ogg



**Prohibited Software**

* Several ways to find prohibited software
* Use the apt logs from above, check for any suspicious packages installed
* comparing initial-status.gz
  + if that file is not on the machine at /var/log/installer/initial-status.gz, get one from a blank machine. TODO GATHER ALL VERSIONS
  + (apt-mark showmanual;sudo zcat initial-status.gz | sed -n '/^Package: / s/^Package: //p')|sort|uniq -u
    - make sure initial-status.gz references the right location
* Remove any games directories
  + rm -rf /usr/lib/games
  + rm -rf /usr/local/games
  + rm -rf /usr/share/games
  + rm -rf /var/games
  + rm -rf /var/lib/games
* List of previously prohibited software
  + 4g8
  + aircrack-ng
  + amass
  + aMule
  + Angry IP Scanner
  + arp-scan
  + autopsy
  + chntpw
  + cmospwd
  + cupp3
  + deluge
  + Doomsday
  + doona
  + dsniff
  + Endless Sky
  + enum4linux
  + Ettercap
  + fcrackzip
  + Freeciv
  + ftpscan
  + Game Conqueror
  + goldeneye
  + hashcat
  + heartbleeder
  + hping3
  + hunt
  + hydra
  + icmpush
  + john the ripper
  + Kismet
  + knocker
  + linuxdcpp
  + ManaPlus
  + medusa
  + Minetest
  + nbtscan
  + nikto
  + Nmap
  + nmapsi4
  + OpenArena
  + OpenRA
  + ophcrack
  + p0f
  + packit
  + Pixel Dungeon
  + pnscan
  + pompem
  + proxychains
  + ptunnel
  + Pumpa
  + rainbowcrack
  + reaver
  + remmina
  + rfdump
  + samdump2
  + scapy
  + skipfish
  + sniffit
  + sqlmap
  + sqlninja
  + strace
  + sucrack
  + tcpblast
  + tcpflow
  + tcpreplay
  + tcpspray
  + themole
  + wapiti
  + Warzone 2100
  + Wireshark
  + xprobe
  + yersinia
  + Zangband
  + zsnes

**Malware**

* View all Running processes
  + ps -ef --forest
  + If it contains a binary with “-lvp 8080” or something of that sort, netcat backdoor
  + If it just has “udevd”, prism backdoor
  + If there is a process with brackets like [loop] that isnt at the top, then its probably a perl backdoor
* Run this command to find any python backdoors
  + sudo ps -aux | grep python
    - You can do the same thing for perl backdoors by replacing python with perl
* Use either sudo netstat -tulpn or sudo lsof -i to scan for listening ports
  + If you found a suspicious program, use the whereis command to find the directory of the program
* Find Web Shells
  + cd /var/www/html
  + grep '((eval.\*(base64\_decode|gzinflate|\$\_))|\$[0O]{4,}|FilesMan|JGF1dGhfc|IIIl|die\(PHP\_OS|posix\_getpwuid|Array\(base64\_decode|document\.write\("\\u00|sh(3(ll|11)))' . -lroE --include=\*.php\*
* Find all SUID files
  + sudo apt-get install git clone
  + sudo git clone https://github.com/Frissi0n/GTFONow.git
    - Compare the output list to [GTFOBins](https://gtfobins.github.io/)
    - Run this command on certain files that was outputted
      * chmod u-s /file/path

**Application Security Settings**

**FTP**

* VSFTPD
  + VSFTPD configurations can be configured in ‘/etc/vsftpd.conf’
  + Add/change the following things:
    - “anonymous\_enable=NO”
    - “chroot\_local\_user=YES”
    - “chroot\_list\_enable=YES”
    - “anon\_world\_readable\_only=YES”
    - “anon\_mkdir\_write\_enable=NO”
    - “anon\_upload\_enable=NO”
    - “anon\_world\_readable\_only=YES”
    - “listen=NO”
    - “listen\_ivp6=NO”
    - “local\_enable=YES”
    - “passive-promiscuous=no”
    - pasv-enable=yes
    - “port-promiscuous=no”
    - port-enable=yes
    - hide\_ids = yes
    - Add logging
      * xferlog\_enable=YES
      * xferlog\_std\_format=NO
      * xferlog\_file=/var/log/vsftpd.log
      * log\_ftp\_protocol=YES
      * debug\_ssl=YES
* PureFTPd
  + PureFTPd can be configured in ‘/etc/pure-ftpd/pure-ftpd.conf’
  + Add/change the following things:
    - "ChrootEveryone yes"
    - "NoAnonymous yes"
    - “AnonymousOnly no”
    - "TLS 2"
    - "MaxClientsNumber 50"
    - "MaxClientsPerIP 3"
    - "MaxIdleTime 10"
    - "LimitRecursion 500 8"
    - "Umask 133:022"
    - “MaxClientsPerIP 2”
    - “VerboseLog yes”
* ProFTPd
  + Pro FTPd can be configured in ‘/etc/proftpd/proftpd.conf’
  + Add/change the following things:
    - “TLSEngine on
    - “TLSLog /var/log/proftpd/tls.log
    - “TLSProtocol SSLv23
    - “TLSRequired on”
    - “TLSVerifyClient off”
    - “TLSOptions NoCertRequest EnableDiags NoSessionReuseRequired”
    - ServerIdent off

**SSH**

* OpenSSH
  + OpenSSH can be configured in ‘/etc/ssh/sshd\_config’
  + Add/change the following things:
    - UsePAM yes
    - AllowTcpForwarding no
    - X11Forwarding no
    - LoginGraceTime 30
    - ClientAliveInterval 300
    - ClientAliveCountMax 0
    - Protocol 2
    - HostBasedAuthentication no
    - PermitEmptyPasswords no
    - StrictModes yes
    - UsePrivilegeSeparation yes
    - PermitRootLogin no
    - PrintLastLog no
    - PermitUserEnvironment no
    - IgnoreRhosts yes
    - AuthenticationMethods publickey
    - PubkeyAuthentication yes
    - MaxAuthTries 3
    - GatewayPorts no

**SMB**

* Samba
  + Samba can be configured in ‘/etc/samba/smb.conf’
  + Add/change the following things:
    - * + ntlm auth = 0
        + smb encrypt = required
        + guest ok = no
        + restrict anonymous = 2
        + min protocol = SMB2
        + server signing = mandatory
        + encrypt passwords = yes
        + obey pam restrictions = yes
        + null passwords = no
        + syslog = 10
        + encrypt passwords = yes
        + usershare allow guests = no
        + guest account = nobody
      * Add/change the following depending on the readme
        + “browseable = no”
        + “read only = yes”
        + “writeable = no”
    - Check inside the share for bad files (DELETE THEM FROM THE SHARE LOCATION NOT THE NETWORK LOCATION)

**HTTP**

* Apache2
  + Apache2 can be configured in ‘/etc/apache2/apache2.conf’
  + Add/change the following:
    - “ServerSignature Off”
    - “FileETag None”
    - “ServerTokens Prod”
    - “TraceEnable Off”
    - “Options -FollowSymLinks”
    - In <Directory /var/www/html>, Add/Change:
      * Options -Indexes
    - In <Directory />
      * “Order Deny,Allow”
      * “​Deny from all”
      * “Options None”
      * “​AllowOverride None”
* Nginx
  + Open nginx.conf
    - Add/edit the following lines in http{}
      * “server\_tokens off”
    - Add/edit the following lines in server{}
      * add\_header X-Frame-Options "SAMEORIGIN"
      * add\_header Strict-Transport-Security max-age=31536000; includeSubdomains; preload";
      * add\_header Content-Security-Policy "default-src 'self' http: https: data: blob: 'unsafe-inline'" always;
      * add\_header X-XSS-Protection "1; mode=block";
      * ssl\_ciphers "EECDH+ECDSA+AESGCM EECDH+aRSA+AESGCM EECDH+ECDSA+SHA384 EECDH+ECDSA+SHA256 EECDH+aRSA+SHA384 EECDH+aRSA+SHA256 EECDH+aRSA+RC4 EECDH EDH+aRSA HIGH !RC4 !aNULL !eNULL !LOW !3DES !MD5 !EXP !PSK !SRP !DSS";
    - Disable TRACE and DELETE HTTP methods
      * Go to the line that has “ if ($request\_method !~” and remove TRACE and DELETE if there
    - Find the line with “ssl\_protocols”
      * Remove any versions that are not “TLSv1.2” or “TLSv1.3”
    - Figure out how to “Inline scripts not allowed by nginx content security policy”

**SQL**

* MySQL
  + MySQL can be configured in ‘/etc/mysql/my.cnf’
  + mysql\_secure\_installation
  + Ensure the line is “user =” is set to “mysql”
  + Add/change the following:
    - “bind-address=localhost”
    - “skip-networking”
    - “local-infile=0”
    - “default\_password\_lifetime=30”
    - “symbolic-links = 0”
* MariaDB
  + Same as MySQL
* PostgreSQL
  + PostgreSQL can be configured at ‘/etc/postgresql/\*/main/postgresql.conf’
  + Add/change the following:
    - “max\_connections = 100”
    - “authentication\_timeout = 1min”
    - “password\_encryption = scram-sha-256”
    - “db\_user\_namespace = off”
    - “ssl = on”
    - ssl\_cert\_file = 'server.crt'
    - ssl\_key\_file = 'server.key'
    - ssl\_prefer\_server\_ciphers = on
    - logging\_collector = on
    - log\_directory = /var/log/postgres
    - log\_hostname = on
    - log\_connections = on
    - log\_disconnections = on
    - log\_error\_verbosity = default
  + pg\_hba.conf
    - hostnossl all all 0.0.0.0/0 reject
    - If a line has trust in it, replace trust with krb5
  + pg\_user\_mappings
    - Make sure there are no insecure user mappings. Delete the line or file if you don’t need any user mappings.
* MongoDB
  + MongoDB can be configured at ‘/etc/mongod.conf’
  + Add/change the following:
    - Under “systemLog:”, add
      * “verbosity: 5”
    - Under “security:”, add
      * authorization: “enabled”

**SMTP**

* Postfix
  + - Postfix can be configured in “/etc/postfix/main.cf”
    - Add/change the following:
      * mail\_owner = postfix
      * smtp\_address\_preference = ipv4
      * inet\_protocols = ipv4
      * html\_directory = no
      * disable\_vrfy\_command=yes
      * inet\_interfaces=loopback-only
      * mynetworks = 0.0.0.0/0
      * myhostname = ubuntu.lan
      * Mydestination = ubuntu.lan, localhost, localhost.lan
      * Smtp\_dns\_support\_level = disabled
      * smtpd\_sasl\_local\_domain = $myhostname
      * smtpd\_helo\_required=yes
      * smtp\_sasl\_auth\_enable = yes
      * smtp\_sasl\_security\_options = noanonymous
      * smtp\_use\_tls = yes
      * smtp\_tls\_loglevel=1
      * broken\_sasl\_auth\_clients = yes
      * smtpd\_sasl\_auth\_enable = yes
      * smtpd\_tls\_received\_header = yes
      * smtp\_tls\_security\_level = may
      * smtpd\_tls\_security\_level = may
      * smtp\_tls\_note\_starttls\_offer = yes
      * smtpd\_data\_restrictions = reject\_unauth\_pipelining
* Dovecot
  + - Dovecot can be configured in “/etc/dovecot/dovecot.conf”
    - Add/change the following:
      * ssl = yes
      * ssl\_verify\_client\_cert = no
      * ssl\_ca =
      * ssl\_protocols = TlSv1.2
      * ssl\_cipher\_list = EECDH+AESGCM:EDH+AESGCM
      * ssl\_prefer\_server\_ciphers
* Roundcube
* Exim
  + - Exim can be configured in “ /etc/exim4/exim4.conf.template”
    - Add/change the following:
      * MAIN\_TLS\_ENABLE = true
      * dc\_eximconfig\_configtype=’satellite’
      * dc\_other\_hostnames=’localhost;domain.com’
      * dc\_local\_interfaces=’127.0.0.1′
      * dc\_readhost=’domain.com’
      * dc\_relay\_domains=”
      * dc\_minimaldns=’false’
      * dc\_relay\_nets=”
      * dc\_smarthost=’smtp.gmail.com::587′
      * CFILEMODE=’644′
      * dc\_use\_split\_config=’true’
      * dc\_hide\_mailname=’true’
      * dc\_mailname\_in\_oh=’true’
      * dc\_localdelivery=’mail\_spool’

**VPN**

* OpenVPN
  + OpenVPN can be configured in “/etc/openvpn/server/server.conf”
  + Add/change the following:
    - ”proto udp”
    - “dev tun”
    - ”chroot jail”
    - ”cipher AES-256-CBC”
    - “auth SHA256”
    - “user nobody”
    - “group nogroup”
    - “verb 9”

**PHP**

* + PHP
    - Find the php.ini file by using the command php -i | grep “php.ini”
      * https://github.com/danehrlich1/very-secure-php-ini