* Change Passwords
* Kill ssh,vnc,telnet,
* Check Running services
  + Displaying All services: service --status-all , chkconfig --list
* Check Users and Groups
  + Finding users in both /etc/passwd & /etc/shadow
  + cut -d: -f1 /etc/passwd
  + cut -d: -f1 /etc/shadow
  + getent group
  + This will print all human users into a file: cut -d: -f1,3 /etc/passwd | egrep ':[0-9]{4}$' | cut -d: -f1 >users.list
* Back up etc and var
  + tar -zcvf archive-name.tar.gz directory-name
  + <https://www.cyberciti.biz/faq/how-do-i-compress-a-whole-linux-or-unix-directory/>
  + Tar -xzvf tar.gz

IP Tables:

iptables:

# if using RHEL/Fedora/CentOS, edit /etc/sysconfig/iptables-config:

# make sure -> IPTABLES\_MODULES\_UNLOAD = no

iptables -P INPUT DROP # by default, drop all incoming connections

iptables -P FORWARD DROP # same for forwarded connections

iptables -P OUTPUT ACCEPT # allow outgoing connections

iptables -L -n -v # status

iptables -A INPUT -m state --state NEW -p tcp --dport 80 -j ACCEPT # eg: allow HTTP

... continue for all necessary services …

service iptables restart # restart the firewall when ready

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 --dport 1514 –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)

Generate SSL certificate:

* Openssl genrsa –des3 –out /etc/pki/tls/certs/example.com.key 1024
* Openssl req –new –key /etc/pki/tls/certs/example.com.key –out /etc/pki/tls/certs/example.csr
* Openssl x509 –req –days 365 –in /etc/pki/tls/certs/example.csr –signkey /etc/pki/tls/certs/example.com.key –out /etc/pki/tls/certs/example.crt
* Modify Apache Config again and add SSL Key
  + <VirtualHost 172.16.25.125:443>
  + SSLEngine on
  + SSLCertificateFile /etc/pki/tls/certs/example.com.crt
  + SSLCertificateKeyFile /etc/pki/tls/certs/example.com.key
  + --Only need for official cert---SSLCertificateChainFile /etc/pki/tls/certs/sf\_budle.crt
  + ServerAdmin blah.blah@example.com
  + ServerName example.com
  + DocumentRoot /var/www/html/example/
  + ErrorLog /var/log/httpd/example.com-error\_log
  + CustomLog /var/log/httpd/example.com-access\_log common
  + </VirtualHost>

Mysql:

Show all databases: **Show databases;**

Select a database: **Use thisdatbase;**

Show databases from selected databases: **SHOW TABLES**

Show columns from within a table: **SHOW COLUMNS FROM city;**

Show columns from specific row: **Show columns from USERS;**

Show contents of a field with a table already selected: SHOW \* FROM USERS;

Create user and give it privileges:

* mysql -u -p
* Create user 'groot' identified by 'gr00tzp@s$w0rd!;
* grant select, insert, update on \*.\* to 'groot';

Secure installation: /usr/bin/mysql\_secure\_installation

Change Mysql password:

* MySQL 5.7.6 and later: ALTER USER 'root'@'localhost' IDENTIFIED BY 'MyNewPass';
* MySQL 5.7.5 and earlier:SET PASSWORD FOR 'root'@'localhost' = PASSWORD('MyNewPass');
* Backup Database
  + mysqldump -u root -p Tutorials > tut\_backup.sql
* Have a look how you can restore your tut\_backup.sql file to the Tutorials database.
  + mysql -u root -p Tutorials < tut\_backup.sql

Tomcat

* locate -b ‘\sample’
* Whereis tomcat5|6
* Su - tomcat
  + Echo $CATALINA\_HOME
  + Echo $CATALINA\_PID
  + $CATALINA\_HOME/bin/startup.sh
  + $CATALINA\_HOME/bin/shutdown.sh

Phpmyadmin:

* <https://www.digitalocean.com/community/tutorials/how-to-install-and-secure-phpmyadmin-on-debian-7>
* http://www.tecmint.com/change-secure-phpmyadmin-login-url-page/

Joomla:

* <https://docs.joomla.org/Security_Checklist>
* [https://www.siteground.com/tutorials/joomla/joomla-security.htm\](https://www.siteground.com/tutorials/joomla/joomla-security.htm/)

LDAP:

* <https://www.howtoforge.com/linux_openldap_setup_server_client>
* <https://wiki.debian.org/LDAP//OpenLDAPSetup>
* Main configuration /etc/ldap/slapd.conf
* <https://www.digitalocean.com/community/tutorials/how-to-change-account-passwords-on-an-openldap-server>
* <http://stackoverflow.com/questions/10553081/change-password-using-ldif-file>

PAM:

* https://access.redhat.com/documentation/en-US/Red\_Hat\_Enterprise\_Linux/6/html/Managing\_Smart\_Cards/PAM\_Configuration\_Files.html

POSTFIX:

* etc/postfix/[main.cf](http://www.postfix.org/postconf.5.html)
* http://www.postfix.org/STANDARD\_CONFIGURATION\_README.html

### DOVECOT:

Find Dovecot configuration file location using:

* + doveconf -n | head -n1

Your configuration file doesn't exist if you installed Dovecot from sources. The config directory should contain a README file pointing to an example configuration, which you can use as your basic configuration. For example:

* cp -r /usr/share/doc/dovecot/example-config/\* /etc/dovecot/

In conf.d/auth-passwdfile.conf.ext you should have:

passdb {

driver = passwd-file

args = scheme=CRYPT username\_format=%u /etc/dovecot/users

}

userdb {

driver = passwd-file

args = username\_format=%u /etc/dovecot/users

}

ROUNDCUBE:

Splunk

**echo "Installing SPLUNK server"**

echo "USER SPLUNK SHOULD ALREADY EXIST"

# Testing on Redhat 7

# I have to extract and then copy the folder because some distros do not have the -C commmand

tar -xzvf ./include/raw/release/splunk6\_64.tgz

cp -Rp ./splunk /opt/splunk

chown -R splunk:splunk /opt/splunk

chmod -R 755 /opt/splunk

/opt/splunk/bin/./splunk start --accept-license

/opt/splunk/bin/./splunk enable boot-start -user splunk

**echo "Installing forwarder"**

echo "USER SPLUNK SHOULD ALREADY EXIST"

# Testing on Redhat 7

# I have to extract and then copy the folder because some distros do not have the -C commmand

tar -xzvf ./include/raw/release/splunkforwarder5\_64.tgz

cp -Rp ./splunkforwarder /opt/splunkforwarder

chown -R splunk:splunk /opt/splunkforwarder

chmod -R 755 /opt/splunkforwarder

/opt/splunkforwarder/bin/./splunk start --accept-license

/opt/splunkforwarder/bin/./splunk enable boot-start -user splunk

echo "First Type in full Host name of Splunk central server (example 192.168.0.1 OR domain.site2.whatever) "

su - splunk -c " whoami;

cd /opt/splunkforwarder/bin;

./splunk add forward-server $\_SHOST:$\_SPORT;

./splunk add monitor /var/log;

./splunk restart;

whoami "

Useful Commands:

Search for string in folder/file: grep -Ri ServerToken \* /etc

Search name or config file or query to search /etc : find /etc/ |grep QUERY

Displaying all Processes: ps -e |grep less

YUM

**echo "YUM will now install only security updates"**

* yum update --security || yum update-minimal --security

**echo "Installing OSSEC Server"**

* wget -q -O – https://www.atomicorp.com/installers/atomic | sh
* yum install ossec-hids ossec-hids-server

**echo "Installing OSSEC Agent**"

* wget -q -O – https://www.atomicorp.com/installers/atomic | sh
* yum install ossec-hids ossec-hids-client

**echo "YUM will install Development Build tools required by programs like OSSEC"**

* yum groupinstall "Development Tools" --setopt=group\_package\_types=mandatory,default,optional

**echo "Type RPM remove name"**

* \_RPMRMPATH=
* read \_RPMRMPATH
* rpm -e "$\_RPMRMPATH"

**echo "Type in local path to RPM"**

* rpm -ivh "$\_RPMPATH"
* #-i to install , v to print verbose (-V to verify) and h to print the h

APT

**echo "What Package Would you like to Filter?"**

* \_DPKG2
* read \_DPKG2
* dpkg --get-selections |grep "$\_DPKG2"

**echo "Now Attempting to Distro Hop"**

* apt-get dist-upgrade

**echo " APT will install Development Build tools required by programs like OSSEC"**

* apt-get install build-essential

PACMAN

* # Temp User needed to compile AUR Packages
* echo "Type In Temp user on system to compile Splunk From AUR"
* echo "USER SPLUNK SHOULD ALREADY EXIST"
* \_TEMPU=
* read \_TEMPU
* su - "$\_TEMPU" -c " curl -L -O https://aur.archlinux.org/cgit/aur.git/snapshot/splunk.tar.gz;
* pwd; tar -xvf splunk.tar.gz;
* pwd; cd ./splunk;
* pwd;
* makepkg -si;
* pwd;"
* pwd
* echo "this"
* cd /home/"$\_TEMPU"/
* pwd
* pacman -U /home/"$\_TEMPU"/splunk/splunk-6.5.1\_f74036626f0c-1-x86\_64.pkg.tar
* su - root -c "chown -R splunk:splunk /opt/splunk"
* su - splunk -c "/opt/splunk/bin/./splunk start --accept-license"
* su - splunk -c "/opt/splunk/bin/./splunk enable boot-start -user splunk"
* }

**function pac\_forwarder**

* echo "Now Configuring Splunk"

* echo "Now Installing the splunk forwarder to /opt/splunkforwarder"
* su - "$\_TEMPU" -c " curl -L -O https://aur.archlinux.org/cgit/aur.git/snapshot/splunkforwarder.tar.gz;
* tar -xvf ./splunkforwarder.tar.gz;
* cd splunkforwarder;
* makepkg -si; "
* pacman -U /home/"$\_TEMPU"/splunkforwarder/splunkforwarder-6.5.1\_f74036626f0c-1-x86\_64.pkg.tar

Important File Paths

/etc/nginx/nginx.conf OR /usr/local/nginx/conf/nginx.conf OR /usr/local/etc/nginx/nginx.conf

Main Configuration File

Main Configuration File

/var/log/httpd/access\_log OR /var/log/httpd/error\_log

Access and Error Logs for the Web Server

/etc/apache2/conf-available/phpmyadmin.conf

Main file for PHPMyAdmin (change path)