Rackspace Compromised Server De-Worming and Lockdown Cheat-Sheet

v2016-10-05_tweeks

1) Log in to our shared, group bastion server:

```
ssh 23.253.57.44
user: <serverXX>
pass: <password>
```

2) From this main server, ssh log in to your serverXX machine:

```
ssh root@serverXX
```

Challenge:

You have been handed a compromised server. It is running a common Internet bot worm that is active and awaiting commands from it's C&C master. Using the command cheat-sheet commands below, evaluate the condition of the serverXX system integrity by identifying any non-standard network profile/port bindings, identifying what processes are responsible, killing those processes, verify they can not re-spawn after a reboot, and that the system binaries (in /bin, /sbin, /usr/bin/ and /usr/sbin/) are all "clean" at a precursory level (i.e. you can trust basic investigatory binaries).

General Strategy:

-**External Network Scan:** Scan machine network profile from outside (nmap)
-**On Server Scan:** Scan machine network profile from on the server (nmap)
-**Check Port Bindings:** Look at port bindings from the server itself (netstat)
-**Examine Configured Services:** Compare bound the running services (chkconfig)
-**Examine Running Services:** (service, ps, etc)
-**Runlevel:** Change from full multi-user to limited multi-user (runlevel, init)
-**Network Lockdown:** Lock down ports (iptables, BE CREAFUL! Can lock yourself out!)
-**Stop Rouge Processes:** Kill any rogue processes (programs with unexpected port bindings)
-**Audit Packages:** Check for any corrupted system binaries, system scripts or config files (rpm)
-**Repair Pacakges:** Repair any corrupted/compromised system binaries, or scripts
-**Reboot Check:** Reboot and check if system is still semi-safe

Port Scan From Outside Your Server (or from the server itself):

```
nmap -sS serverXX       # general /etc/services stealth port scan
nmap -sS -p 1-5000      # specific port range scan (scans ports from 1-5000)
```

Working with SysV-init Based Services (not systemd based systems):

```
#Check/list all services:
ls /etc/init.d/
runlevel                        # list our previous and current run levels
chkconfig --list                # list all services
chkconfig --list | grep 3:on    # only list services configured "on" for RL 3
service <servicename> [start|stop|status|restart]
                                # ex: service httpd restart
init 2                          # change to runlevel 2 (limited multiuser)
init 3                          # change back to standard, headless runlevel
```

Examining/Working with Networking (from on your server):

```
ip address show          # show configured IP addresses (the new way)
ifconfig                 # show configured IP addresses (the old way)
netstat -ant             # see [a]ll [n]umerical [t]cp
netstant -antp           # same as above + show process names bound to ports
iptables -L              # list human readable listing of iptables ACLs chains
/etc/sysconfig/iptables       # the iptables ACL tabble/chain listings
service iptables restart      # to restart (and reload config) of iptables
```

WARNING: It is very easy to lock yourself out with iptables.
Ask for help if unsure BEFORE restarting iptables.
Especially when you do not have physical access to
a system's console.

Working with Packages:

```
rpm -qa             # list all installed packages
yum list            # list all packages available in the configured package repos
rpm -qi <package>   # show info about a given installed packages from local package
yum info <package>  # ^^ same as above ^^, but gets info from the repo meta data
rpm -ql <package>   # list all files within a given packages
rpm -Uvh <package>  # install/upgrade a local or remote(URL) rpm file (no autodeps)
yum install <pkg>   # install package from repo & resolve dependencies
rpm -V  <package>   # verify a package (check file MD5 fingerprints) on disk vs dbase
rpm -Va             # verify all package file fingerprints
rpm -Va | grep ^..5 # show files that have changed (fingerprint) since install time
rpm -qf <file>      # list the package a file belongs to
yum reinstall <package>   # reinstall same (clean) version of installed package
```

Process Listing & Manipulation:

```
ps -auxw                # print processes, [a]ll, [u]users, [x]BSDish, [w]ide
kill <PID>              # kill a specific process ID number
killall <process-name>  # kill all processes with a given name
pkill <process-name>    # kill all process names by regular-expression
```

General Server Control:

```
reboot                       # gracefully reboots the server
poweroff                # graceful power down
```