Linux Hardening

By RJ

Users, Groups, and Permissions

Principle of Least Privilege

- A user should have the minimum amount of privilege required for completing their activities
- If an unprivileged account is compromised, attackers can’t do much with it
- Create users and groups with permissions for specific purposes

Listing Users

- `w` command shows lots of info about who is logged in

```
rj@DawgCTFPractice:~$ w
 16:11:08 up 2 days, 17:46,  3 users,  load average: 0.00, 0.00, 0.00
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
rj       tty1     -                16:07    3:40   0.17s  0.06s -bash
root     pts/0    130.85.59.149    16:11    4.00s  0.03s  0.03s -bash
rj       pts/1    130.85.59.149    16:00    1.00s  0.05s  0.00s w
```

- TTY - Terminal given to directly connected user
- PTS - Terminal given to remotely connected user (i.e. ssh or telnet)

/etc/passwd, /etc/group, /etc/shadow

- /etc/passwd stores info about each user
  - UID, GID, home dir, shell
  - /bin/false vs /usr/sbin/nologin
- /etc/group stores info about each group
  - GID and users who belong to the group
- /etc/shadow stores password hashes
  - Hash type, salt, password hash

/etc/sudoers

- Specifies who can run commands that require root privileges
- Format: user (host)=(user:group) commands

Linux Permissions

```
rj@DawgCTFPractice:~$ ls -la example.sh
-rwxrw-r-- 1 rj rj 0 Sep 15 16:33 example.sh
```

- Owner can read, write, and execute
- Members in owner's group can read and write
- All other users can read

Changing owner, group, and permissions

- chown [user] [path]
- chgrp [group] [path]
- chmod [permissions] [path]

Attack Surface

Attack Surface

- Attack surface: combination of all methods an attacker could use to gain access to a system
- Need to be aware of what services services are running on a system and how to secure them
- Need to know what is coming in and out of network

Listing Processes

- ps -ef | less -S

```
root     29814 29438  0 Sep16 ?        00:02:14 docker-gen -watch -notify /app/signal_le_service -
root     30139   991  0 Sep16 ?        00:00:00 /usr/bin/docker-proxy -proto tcp -host-ip 0.0.0.0
root     30166   991  0 Sep16 ?        00:00:00 /usr/bin/docker-proxy -proto tcp -host-ip 0.0.0.0
root     30187   974  0 Sep16 ?        00:00:02 containerd-shim -namespace moby -workdir /var/lib/
root     30200   974  0 Sep16 ?        00:00:02 containerd-shim -namespace moby -workdir /var/lib/
rj       30241 30187  0 Sep16 ?        00:00:00 /bin/sh -c socat -T10 -dd TCP-LISTEN:5000,reuseadd
rj       30253 30200  0 Sep16 ?        00:00:00 /bin/sh -c socat -T10 TCP-LISTEN:5000,reuseaddr,fo
rj       30420 30241  0 Sep16 ?        00:00:00 socat -T10 -dd TCP-LISTEN:5000,reuseaddr,fork EXEC
rj       30463 30253  0 Sep16 ?        00:00:00 socat -T10 TCP-LISTEN:5000,reuseaddr,fork EXEC:/ho
root     30644   991  0 Sep16 ?        00:00:00 /usr/bin/docker-proxy -proto tcp -host-ip 0.0.0.0
root     30652   974  0 Sep16 ?        00:00:02 containerd-shim -namespace moby -workdir /var/lib/
```

Listing Listening Network Connections

- netstat -tulpn

```
root@DawgCTFPractice:~# netstat -tulpn
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      727/systemd-resolve
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      8296/sshd
tcp        0      0 0.0.0.0:13370           0.0.0.0:*               LISTEN      2188/python3
tcp        0      0 0.0.0.0:13371           0.0.0.0:*               LISTEN      2189/python3
tcp        0      0 0.0.0.0:13372           0.0.0.0:*               LISTEN      2190/python3
tcp        0      0 0.0.0.0:13373           0.0.0.0:*               LISTEN      2191/python3
tcp        0      0 0.0.0.0:13374           0.0.0.0:*               LISTEN      2192/python3
tcp6       0      0 :::4000                 :::*                    LISTEN      11608/docker-proxy
tcp6       0      0 :::3500                 :::*                    LISTEN      30166/docker-proxy
tcp6       0      0 :::80                   :::*                    LISTEN      1944/docker-proxy
tcp6       0      0 :::8081                 :::*                    LISTEN      30644/docker-proxy
tcp6       0      0 :::22                   :::*                    LISTEN      8296/sshd
tcp6       0      0 :::3000                 :::*                    LISTEN      30139/docker-proxy
tcp6       0      0 :::443                  :::*                    LISTEN      29278/docker-proxy
```

Hardening Services

- Services in Linux are highly configurable
  - Often come with a configuration file

- Usually many security configuration options available - research them and configure the service properly

- Usually can just google “securing whatever service” and someone’s written a guide about it

Patching Service Vulnerabilities

- Check for out of date services and make sure you are running the latest version!

- Research CVEs for services and apply appropriate mitigations

Firewall Rules

- Will go into this in much more depth in a couple weeks
- Can set rules on what traffic is allowed in and out of a computer or network
- Show firewall rules using iptables -nvL

SSH

- Protocol for remotely accessing a Linux system

- Pay very close attention to how SSH is configured!
  - /etc/ssh/sshd_config

History and Logging

- Bash history of a user is stored in .bash_history file in their home directory

- Logs are stored in /var/log
  - /var/log/auth.log shows all authentication attempts
  - Services often have their own log files