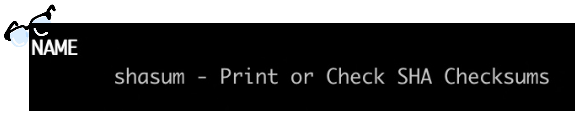
Born2beRoot Defense Checklist

Note: If something does not work as expected or is not clearly explained, stop evaluation. When you need help with checking something, the student should be able to help you.

Preliminary tests

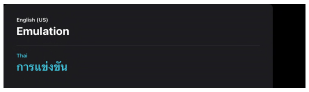
- Git repo cloned successfully.



NAME
       shasum - Print or Check SHA Checksums

General instructions

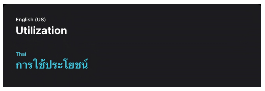
- Git repo contains a signature.txt file.
- Check the signature against the students ".vdi" file, make sure it's identical.
- Create a snapshot && open VM.



English (US)
Emulation
Thai
การแข่งขัน

Mandatory Part (Questions for the student)

- *How does a virtual machine work and what is its purpose?*
  A virtual machine (VM) is a software-based emulation of a physical computer that runs an operating system and applications. Its purpose is to provide a way to run multiple operating systems on a single physical machine, enabling isolation, flexibility, and efficient resource utilization.



English (US)
Utilization
Thai
การใช้ประโยชน์

- *The basic differences between Rocky and Debian?*
  Rocky Linux is a CentOS replacement geared toward enterprise use, known for its stability, while Debian is a versatile distribution suitable for a wide range of applications, offering a flexible release cycle and extensive software repositories.
- *Their choice of operating system?*
  Debian
- ~~*If CentOS: what SELinux and DNF are.*~~
- *If Debian: the difference between aptitude, apt and what APPArmor is.*
  Aptitude is a terminal-based package manager that provides a text-based interface for package management. It offers features like dependency resolution and package tracking.

  Apt (or APT) is the package management command-line tool that interfaces with the Advanced Package Tool (APT) libraries. It allows you to install, upgrade, and manage packages on your system.

  AppArmor is a Linux security module that provides mandatory access control for programs and system processes. It restricts the capabilities of applications to enhance system security by defining what resources they can access and modify.
- During the defense, a script must display all information every 10 minutes. Its operation will be checked in detail later.
- All explanations are satisfactory (else evaluation stops here).

Simple setup

- Ensure that the machine does not have a graphical environment at launch.
- Connect to VM as a created user (which isn't a root)
- Ensure the password follows the required policy (2 days min, 7, 30 days max).

  ```
  sudo vim /etc/login.defs
  sudo chage -l username
  ```
- Evaluator checks UFW service is started.

  ```
  sudo ufw status                    //look for status: active
  ```
- Evaluator checks SSH service is started.

  ```
  sudo systemctl status ssh
  ```
- Evaluator checks the chosen operating system (Debian or Rocky).

  ```
  lsb_release -a || cat /etc/os-release
  ```

User

- The subject requests that a user with the login of the student being evaluated is present on the virtual machine. Check that it has been added and that it belongs to "sudo" and "user42" groups.

```
getent group sudo
getent group user42
```

Password policy check:

- Create new user (e.g. user42).

```
sudo adduser new_username
```

- Assign a password of your choice, respecting subject rules.

```
getent group sudo
```

- Explanation from student explaining how to implement the password policy.

  First type sudo apt-get install libpam-pwquality to install Password Quality Checking Library

  1. Then type sudo vim /etc/pam.d/common-password
  2. Find this line. password requisite pam_deny.so
  3. Add this to the end of that line `minlen=10 ucredit=-1 dcredit=-1 maxrepeat=3 reject_username difok=7 enforce_for_root`

     minlen=10: Requires passwords to have a minimum length of 10 characters.
     ucredit=-1: Mandates the inclusion of at least one uppercase letter (ucredit) in the password (the -1 means it's required).
     dcredit=-1: Requires at least one digit (dcredit) in the password (the -1 means it's required).
     maxrepeat=3: Limits the repetition of the same character in the password to a maximum of 3 times.

reject_username: Prevents the use of the username as part of the password.
difok=7: Allows passwords to differ from the previous one by at least 7 characters.
enforce_for_root: Enforces these requirements for the root (superuser) account as well.

Save and Exit Vim

- Next type in your Virtual Machine sudo vim /etc/login.defs
- Find this part PASS_MAX_DAYS 9999 PASS_MIN_DAYS 0 PASS_WARN_AGE 7
- Change that part to PASS_MAX_DAYS 30 and PASS_MIN_DAYS 2 keep PASS_WARN_AGE 7 as the same
- Lastly type sudo reboot to reboot the change affects

- Normally there should be one or two modified files. If there is any problem, the evaluation stops here.
- With the new user, ask the student to create a group named “evaluating” and assign it to the new user.

sudo groupadd evaluating
sudo usermod -aG evaluating your_new_username

- Check if the new user belongs to the “evaluating” group.

getent group evaluating

- Ask the student to explain advantages of the password policy (beyond the fact that it is required for the project)

  Security, Data Protection, Reduced Risk

- Ask the student the advantages/disadvantages of the policy implementation.

  Advantages:
  Enhanced Security: Strong policies improve security.
  User Education: Promotes user awareness about password hygiene.
  Compliance: Ensures adherence to regulatory requirements.
  Reduced Vulnerabilities: Decreases the likelihood of password-related vulnerabilities.

  Disadvantages:
  User Frustration: Strict policies may lead to user frustration or forgotten passwords.
  Support Overhead: Users may require assistance with password resets.
  Complexity: Overly complex policies might be hard to enforce and understand.
  Resistance: Users may resist policy changes, leading to non-compliance.
  Initial Costs: Implementing and enforcing policies may require additional resources.

Hostname and partitions

- Check the hostname of the machine is correctly formatted as follows: login42 (login of the student being evaluated).

```
hostnamectl
```

- Modify this hostname by replacing the login with yours, then restart VM.

```
sudo hostnamectl set-hostname new_hostname
sudo reboot
```

Note: If on restart, the hostname has not been updated, the evaluation stops here.

- Restore the machine to the original hostname, then restart VM.

```
sudo hostnamectl set-hostname new_hostname
sudo reboot
```

- Ask the student being evaluated how to view the partitions for the VM.

```
lsblk
```

- Compare the output with the example given in the subject (if there are bonuses, refer to the bonus example).
- Ask the student for a brief explanation of LVM and how it works.
  LVM, or Logical Volume Manager, is a technology used in Linux and Unix-like operating systems to manage storage devices and create flexible storage solutions. It allows for the creation of logical volumes, which are virtual storage entities that can span across multiple physical disks

SUDO

- Check that the “sudo” program is properly installed on the virtual machine.

  ```
  dpkg -l | grep sudo
  ```

- The student being evaluated shows assigning a new user to the “sudo” group.

  ```
  sudo usermod -aG sudo <username>
  ```

- The subject imposes strict rules for sudo. The student being evaluated must explain the value and operation of sudo using examples of their choice.

```
sudo visudo ls
```

- Second step, must show the implementation of the rules imposed by the subject.
- Verify the “/var/log/sudo/” folder exists and has at least one file.
- Run a command via sudo. See if the file(s) in the “/var/log/sudo/” folder have been updated.

UFW

- Check the “UFW” program is properly installed on the VM and works properly.

```
sudo ufw status numbered
```

- Ask the student for a basic explanation of UFW and the value of using it.
- List the active rules in UFW. A rule must exist for port 4242.
- Add a new rule to open port 8080. Check that this one has been added by listing the active rules.

```
sudo ufw allow 8080
```

- Delete this new rule with the help of the student being evaluated.

```
sudo ufw delete 4
sudo ufw delete 2
```

SSH

- Check that the SSH service is properly installed on the VM, and is working properly.

```
sudo service ssh status                    //check if its active and port 4242
```

- Ask the student for an explanation of what SSH is and the value of using it. *(answer: secure shell, allows 2 computers to securely talk to each other)*
- Verify that the SSH service only uses port 4242.
- Ask the student to help you use SSH in order to log in with the newly created user. To do this, you can use a key or simple password, depending on the student being evaluated.

```
ssh new_user@127.0.0.1 -p 4242
```

- Make sure you cannot use SSH with the “root” user as stated in the subject.

```
ssh amusso-g42@127.0.0.1 -p 4242           //should come up as permission denied
```

Script Monitoring (questions for the student)

- Ask the student how their script works and see their code for it.

Script inputted in the monitoring .sh file to display system information

```
cd /usr/local/bin && vim monitoring.sh
```

- *What is “cron”?*
  "Cron" is a time-based job scheduler in Unix-like operating systems, including Linux. It allows users to schedule and automate tasks or scripts to run at specified intervals or times. These scheduled tasks are referred to as "cron jobs."
- *How does the script run every 10 minutes from when the server starts?*
  Type sudo crontab -u root -e to open the crontab and add the rule
  *Lastly at the end of the crontab, type the following */10 * * * * /usr/local/bin/monitoring.sh this means that every 10 mins, this script will show*

arc=$(uname -a): Retrieves system architecture information using the uname -a command.

pcpu=$(grep "physical id" /proc/cpuinfo | sort | uniq | wc -l): Counts the number of physical CPUs on the system by examining the CPU information in /proc/cpuinfo.

vcpu=$(grep "^processor" /proc/cpuinfo | wc -l): Counts the total number of virtual CPUs (processors) on the system by examining CPU information.

fram=$(free -m | awk '$1 == "Mem:" {print $2}'): Retrieves the total RAM (in MB) using the "free" command and extracts it from the output.

uram=$(free -m | awk '$1 == "Mem:" {print $3}'): Retrieves the used RAM (in MB) using the "free" command.

pram=$(free | awk '$1 == "Mem:" {printf("%.2f"), $3/$2*100}'): Calculates and displays the percentage of used RAM.

fdisk=$(df -BG | grep '^/dev/' | grep -v '/boot$' | awk '{ft += $2} END {print ft}'): Retrieves the total disk space in GB by examining the output of the "df" command.

udisk=$(df -BM | grep '^/dev/' | grep -v '/boot$' | awk '{ut += $3} END {print ut}'): Retrieves the used disk space in MB.

pdisk=$(df -BM | grep '^/dev/' | grep -v '/boot$' | awk '{ut += $3} {ft+= $2} END {printf("%d"), ut/ft*100}'): Calculates and displays the percentage of used disk space.

cpul=$(top -bn1 | grep '^%Cpu' | cut -c 9- | xargs | awk '{printf("%.1f%%"), $1 + $3}'): Retrieves the CPU load percentage.

lb=$(who -b | awk '$1 == "system" {print $3 " " $4}'): Displays the date and time of the last system boot.

lvmu=$(if [$(lsblk | grep "lvm" | wc -l) -eq 0]; then echo no; else echo yes; fi): Checks if the system uses LVM (Logical Volume Manager) and displays "yes" or "no" accordingly.

ctcp=$(ss -neopt state established | wc -l): Counts the number of established TCP connections.

ulog=$(users | wc -w): Counts the number of logged-in users.

ip=$(hostname -I): Retrieves the system's IP address.

mac=$(ip link show | grep "ether" | awk '{print $2}'): Retrieves the MAC (Ethernet) address of the system.

cmds=$(journalctl _COMM=sudo | grep COMMAND | wc -l): Counts the number of sudo commands in the system's journal logs.

wall " ... ": Uses the "wall" command to broadcast the gathered system information to all logged-in users.