Shell Scripting

Due 14 March 2024

يعد الغش مخالفة أكاديمية وفقا للوائح والقوانين المعمول بها في جامعة قطر، وقد تصل عقوبة هذه المخالفة في بعض الحالات إلى الفصل النهائي من الجامعة وعلى الطلاب تجنب القيام أو المشاركة في أي عمل يخالف ميثاق النزاهة الاكاديمية وإجراءات الاختبارات المعمول بها بجامعة قطر.

Cheating is an academic violation according to Qatar University rules and regulations, and in some cases, it may result in final dismissal from the University. Students should not under any circumstances commit or participate in any cheating attempt or any act that violates the student code of conduct.

Project Objectives:

1. **Server and Client Configuration on VM environments:** Learn the basics of server setup, including web server installation, SSH, and SFTP, along with client machine configuration for network operations.
2. **Security and User Authentication:** Implement security measures by managing user access and permissions, ensuring that only authorized users can perform specific tasks.
3. **Network Services Automation:** Automate common network engineer tasks, including network configuration checks, system monitoring, and file management through shell scripting.
4. **Practical Problem Solving:** Apply theoretical knowledge to solve a practical problem faced by network engineers, emphasizing the automation of repetitive tasks to improve efficiency and accuracy.

Environment Setup:

- **Virtual Machines:** Use Ubuntu Desktop or Fedora Workstation for creating three VMs: VM1 (server), VM2 (client), and VM3 (client for testing automation).

Tasks:

Task 1: Setup

1. **Server Administrator Tasks (VM1):**

- Create local user accounts for each client on the server.
- Install and enable a web server (Apache or NGINX).
- Install and enable SSHD.
- Configure SFTP for file transfer operations.

2. **Client Machine Setup (VM2):**

- Install and enable SSH (this is installed, by default, maybe validate it instead?).

Task 2: Configuration

1. **Server Configuration (VM1):**

- **Webserver Setup:**
  - Create a welcome page with the message: "Welcome to Operating Systems Lab".
  - Implement user authentication for web page access.
  - Log all access attempts.

- **SSH and SFTP Configuration:**
  - Enable SSH access for clients using local account credentials, that is, configure SSHD server such that clients can access the webserver via their local accounts, using username and password.
  - Set up SFTP for file uploads/downloads.

- **DNS Configuration:**
  - Configure local DNS to use 9.9.9.9 or 149.112.112.112, *Quad9's open DNS recursive service for free security and high privacy***,** as the default DNS servers.

- **Manage Unsuccessful Attempts Log**
  - Consolidate all unsuccessful attempt logs, that are synchronized from the clients, in `unsuccessful_attempts.log`. *Run a background task to consolidate the logs*.
  - Automatically erase this log after one week, from consolidation.

2. **Client Configuration (VM2):** Develop a main script (`main.sh`) to manage user credentials and perform the following:

   Step 1: Check User Group Membership

   - If the current user is part of the "**clients**" group, proceed with script execution.

- If not, prompt the user to obtain superuser privileges (without using the root account) to add them to the “clients” group (create the group if it does not exist) and then run the scripts.

Step 2: Log Invalid Attempts

- Log every invalid attempt to the server, using SSH, with the user's name and timestamp in `invalid_attempts.log`. *A script that receives a username and password then tries to login to the server using those credentials.*
- Limit to 3 attempts before taking further action.

Step 3: Handle Excessive Invalid Attempts

- Upon exceeding the maximum attempts, display "Unauthorized user!".
- Copy `client_timestamp_invalid_attempts.log` using the `rsync` tool to the server.
- Schedule a user logout from the desktop session after one minute. *gnome-session-quit --no-prompt*

Step 4: Verification

Create a `verify.sh` script that performs the following:

a) Create a user named "techuser" with a password.
b) Attempt running `main.sh` under "techuser".
c) Create a new group named "admins".
d) Add "admins" to the sudoers file (`/etc/sudoers`).
e) Include "techuser" in the "admins" group.
f) Try executing `main.sh` again as "techuser".

Task 3: Automation

Develop and execute bash scripts for both server and client VMs to automate the setup and configuration tasks detailed in Tasks 1 and 2. *Expand on the filenames and structure*.

Task 4: Network Connectivity Testing and Troubleshooting Task

To ensure proper network connectivity between the one server VM (VM1) and two client VMs (VM2 and VM3) you setup, now your task is testing connectivity between these machines and automate pinging from each VM to others.

1. Identify the IP addresses of VM1 (server), VM2 and VM3 (clients).
2. Ensure SSH access is configured properly as you might need to run this script remotely or transfer it between machines using SFTP for execution.
3. Execute the script: Use `./meshping.sh "target_IPs"` to test connectivity, using `ping`, from VM1 to VM2 and VM3, and vice versa. This script must run **3** times in total. *A timeout must be used.*

- If the target VM(s) is not responding, run a second script, `traceroute.sh`, which checks again the connectivity using trace route, and reboots the machine in case the target cannot be reached.
- All information is saved in a log file (network.log) and displayed on the screen.
- Otherwise, it should show the current date in the following format: "2024-02-14 10:03:04" along with this message "Connectivity with $target_IP is ok".

There is a ½ bonus mark if you can demonstrate your project with the virtual machines running on different *physical* machines.

Instructions & Deliverables: Please read carefully

1- In MS-Word document submit the following:
   - Cover page includes necessary details of the group members (Student Name, Student ID)
   - In a table format, specify each group member tasks and the contribution percentage into the project.

2- Submit your scripts with the MS-Word document on the Blackboard in one zipped file no later than **the below due date**

   Due 14 March 2024

   Call the Zip file (StuentNAME1_StudentNAME2_StudentNAME3_StudentNAME4)

3- Copying and/or plagiarism (-100%) which includes:
   Inappropriate interaction with any other student, outside agency, website, or software that generates assessment responses.
4- Shell Script should be error free (Errors) (-25%).
5- In case of late submission, (-10%) for each day of delay (Max 3 days delay).
6- A group of three to four students can work on the project.
7- Team members are required to meet regularly for discussion and workload distribution.
8- It is the right of the instructor to use any way of testing the student in the discussion and demo session, and according to that in some cases (100%) graded project may be down to (-100%) graded project.
9- Discussion & Demo 50%. + Student Work 50%.