Remote Control Automation Script

STUDENT NAME: Tomer Deri

Unit: TMagen773637

Student Code: s5

Program Code: nx201

Script Explanation

This bash script automates remote reconnaissance tasks using a secure and anonymous connection through the Tor network. It installs required tools, verifies anonymity via Nipe, and connects to a remote server over SSH to perform a WHOIS lookup and an Nmap scan on a user-defined target. All results are downloaded locally and logged automatically.

Overview of Functions

1. Defines color variables for structured terminal output.

```
# ===== Colors =====
RED="\e[31m"
GREEN="\e[32m"
YELLOW="\e[33m"
BLUE="\e[34m"
RESET="\e[0m"

info() { echo -e "${YELLOW}[INFO]${RESET} $1"; }
success() { echo -e "${GREEN}[OK]${RESET} $1"; }
error() { echo -e "${RED}[ERROR]${RESET} $1"; }
```

2. Creates an output folder and log file to store scan results.

```
# ===== Create output directory =====
mkdir -p output
LOG_FILE="output/audit_log_$(date +%Y%m%d_%H%M%S).txt"
```

3. Installs essential tools like sshpass, nmap, whois, git, curl, and Nipe.

```
# ===== Install Required Tools =====
install_tools() {
    info "Checking and installing required tools..."
    packages=(nmap whois sshpass git curl)
    for pkg in "${packages[@]}"; do
        if ! command -v $pkg &>/dev/null; then
            info "Installing $pkg..."
            sudo apt-get install -y $pkg
        else
            success "$pkg already installed."
        fi
    done
```

4. Clones and sets up the Nipe repository if not already present.

```
# ===== Install and Setup Nipe =====
install_nipe() {
    if [ ! -d nipe ]; then
        info "Cloning Nipe repository..."
        git clone https://github.com/htrgouvea/nipe.git
        cd nipe
        sudo cpan install Try::Tiny Config::Simple JSON
        sudo perl nipe.pl install
        cd ..
    else
        success "Nipe already exists."
    fi
}
```

5. Starts Nipe and verifies anonymity using check.torproject.org.

```
# ===== Start Nipe and Check Anonymity =====
check_anonymity() {
    cd nipe
    sudo perl nipe.pl start
    sudo perl nipe.pl restart
    sleep 5

    tor_status=$(curl -s https://check.torproject.org | grep -o "Congratulations. This browser is configured to use Tor")
    cat
    if [[ "$tor_status" != "" ]]; then
        country=$(curl -s https://ipinfo.io/country 2>/dev/null)
        if [[ "$country" == *"<html>"* ]]; then
            country="Unknown (Blocked by ipinfo.io)"
        fi
        success "You are now anonymous. Spoofed country: $country"
    else
        error "Anonymity check failed. You are NOT using Tor."
        exit 1
    fi
    cd ..
}
```

6. Prompts the user to enter SSH details and a scan target.

```
# ===== Connect and Execute Commands on Remote Server =====
remote_operations() {
    echo ""
    read -p "Enter remote server IP: " server_ip
    read -p "Enter remote username: " user
    read -s -p "Enter remote password: " password
    echo ""
    read -p "Enter target address to scan: " target

    timestamp=$(date +%Y%m%d_%H%M%S)
```

7. Connects to the remote server via SSH and performs the whois and nmap scan.

```
    info "Connecting to remote server and executing commands..."
    sshpass -p "$password" ssh -o StrictHostKeyChecking=no $user@$server_ip << EOF
        echo "[REMOTE] Server Uptime:"
        uptime
        echo "[REMOTE] Server Country:"
        curl -s ipinfo.io/country

        echo "[REMOTE] Performing WHOIS lookup on $target..."
        whois $target > whois_$target.txt

        echo "[REMOTE] Performing Nmap scan on $target..."
        nmap -Pn $target -oN nmap_$target.txt
EOF
```

8. Downloads result files from the remote server.

```
info "Downloading result files from remote server..."
sshpass -p "$password" scp $user@$server_ip:whois_$target.txt output/whois_$target_$timestamp.txt
sshpass -p "$password" scp $user@$server_ip:nmap_$target.txt output/nmap_$target_$timestamp.txt

success "Files downloaded to output/"
```

9. Saves all collected data into a log file for auditing.

```
    # Logging
    {
        echo "==================== AUDIT LOG ===================="
        echo "Scan Timestamp: $(date)"
        echo "Target: $target"
        echo "Remote Server IP: $server_ip"
        echo "Spoofed Country: $country"
        echo "Remote Results:"
        echo "- Whois: whois_$target_$timestamp.txt"
        echo "- Nmap:  nmap_$target_$timestamp.txt"
        echo "=================================================="
    } >> "$LOG_FILE"

    success "Audit log created: $LOG_FILE"
}
```

Execution Flow

1. Clear the screen and display the script banner.

2. Install required tools silently if missing.

3. Set up and verify Tor-based anonymity with Nipe.

4. Ask for user input to define remote connection and scan target.

5. Perform reconnaissance on the remote server and retrieve results.

6. Store logs and downloaded files in the "output" directory.

Output Files

- whois_<target>_<timestamp>.txt: WHOIS result from the remote scan.

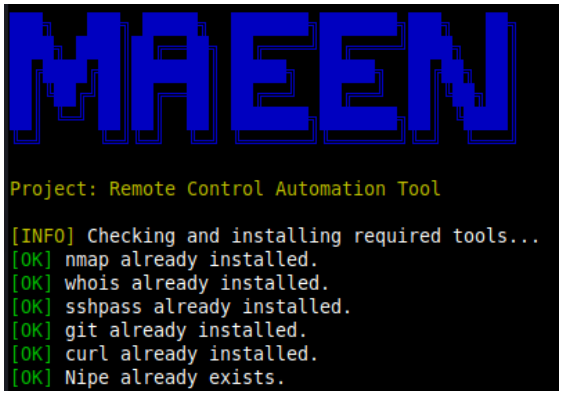
- nmap_<target>_<timestamp>.txt: Port scan result from the remote scan.

- audit_log_<timestamp>.txt: Log containing all metadata and actions performed.

Script Execution Screenshots

Below you may insert screenshots showing each step of the script execution, including:

- The script running in the terminal.



- The spoofed IP or spoofed country verification.

```
[OK] You are now anonymous. Spoofed country: Unknown (Blocked by ipinfo.io)
```

- The SSH connection to the remote server.

```
Enter target address to scan: scanme.nmap.org
[INFO] Connecting to remote server and executing commands...
Pseudo-terminal will not be allocated because stdin is not a terminal.
Warning: Permanently added '192.168.153.129' (ED25519) to the list of known hosts.
Linux kali 6.12.25-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.25-1kali1 (2025-04-30) x86_64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
[REMOTE] Server Uptime:
 10:29:48 up  8:00,  2 users,  load average: 0.30, 0.26, 0.20
[REMOTE] Server Country:
US
[REMOTE] Performing WHOIS lookup on scanme.nmap.org...
[REMOTE] Performing Nmap scan on scanme.nmap.org...
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-23 10:29 EDT
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up.
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
All 1000 scanned ports on scanme.nmap.org (45.33.32.156) are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

Nmap done: 1 IP address (1 host up) scanned in 203.44 seconds
```

- The content of the output folder.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ ls output/

audit_log_20250623_102701.txt  nmap_20250623_102947.txt  whois_20250623_102947.txt
```

- A sample from each result file and the audit log.

```
┌──(kali㉿kali)-[~/Desktop]
└─$ cat output/audit_log_*.txt

==================== AUDIT LOG ====================
Scan Timestamp: Mon Jun 23 10:33:13 AM EDT 2025
Target: scanme.nmap.org
Remote Server IP: 192.168.153.129
Spoofed Country: Unknown (Blocked by ipinfo.io)
Remote Results:
- Whois: whois_20250623_102947.txt
- Nmap:  nmap_20250623_102947.txt
===================================================
```

```
┌──(kali㉿kali)-[~/Desktop]
└─$ cat output/nmap_*.txt

# Nmap 7.95 scan initiated Mon Jun 23 10:29:49 2025 as: /usr/lib/nmap/nmap --privileged -Pn -oN nmap_scanme.nmap.org.txt scanme.nmap.o
rg
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up.
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
All 1000 scanned ports on scanme.nmap.org (45.33.32.156) are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

# Nmap done at Mon Jun 23 10:33:13 2025 -- 1 IP address (1 host up) scanned in 203.44 seconds
```