



# PROJECT: REMOTE CONTROL

NETWORK RESEARCH

[JMagen773630.s17.nx201.pdf](#)

Network Research Project  
Nissim Atar  
s17  
JMagen773630

# 1. Installations and Anonymity Check

The screenshot displays a Kali Linux virtual machine environment. The main window shows a terminal with a script for installing and checking anonymity. The script is titled "1.1 Install the needed applications." and includes functions for installing Sshpass, Nmap, Torify, and Nmap. It also includes a function for checking the network connection and displaying the user's IP address. The script is written in Bash and includes comments in Chinese. The terminal output shows the script running successfully, with messages indicating the installation of Sshpass, Nmap, and Torify, and the successful execution of the anonymity check.

The right window shows a web browser displaying the project structure. The page title is "NETWORK RESEARCH | PROJECT: REMOTE CONTROL". The page content includes a "Project Structure" section with the following items:

- 1. Installations and Anonymity Check
  - 1.1 Install the needed applications.
  - 1.2 If the applications are already installed, don't install them again.
  - 1.3 Check if the network connection is anonymous; if not, alert the user and exit.
  - 1.4 If the network connection is anonymous, display the spoofed country name.
  - 1.5 Allow the user to specify the address to scan via remote server; save into a variable.
- 2. Automatically Connect and Execute Commands on the Remote Server via SSH
  - 2.1 Display the details of the remote server (country, IP, and Uptime).
  - 2.2 Get the remote server to check the Whois of the given address.
  - 2.3 Get the remote server to scan for open ports on the given address.
- 3. Results
  - 3.1 Save the Whois and Nmap data into files on the local computer.
  - 3.2 Create a log and audit your data collecting.
- 4. Creativity

The "General" section includes the following items:

- Suggested tools: Sshpass, Nmap, Torify, Nmap, Whois.
- Everything other than the user input should be automated.
- Use functions.

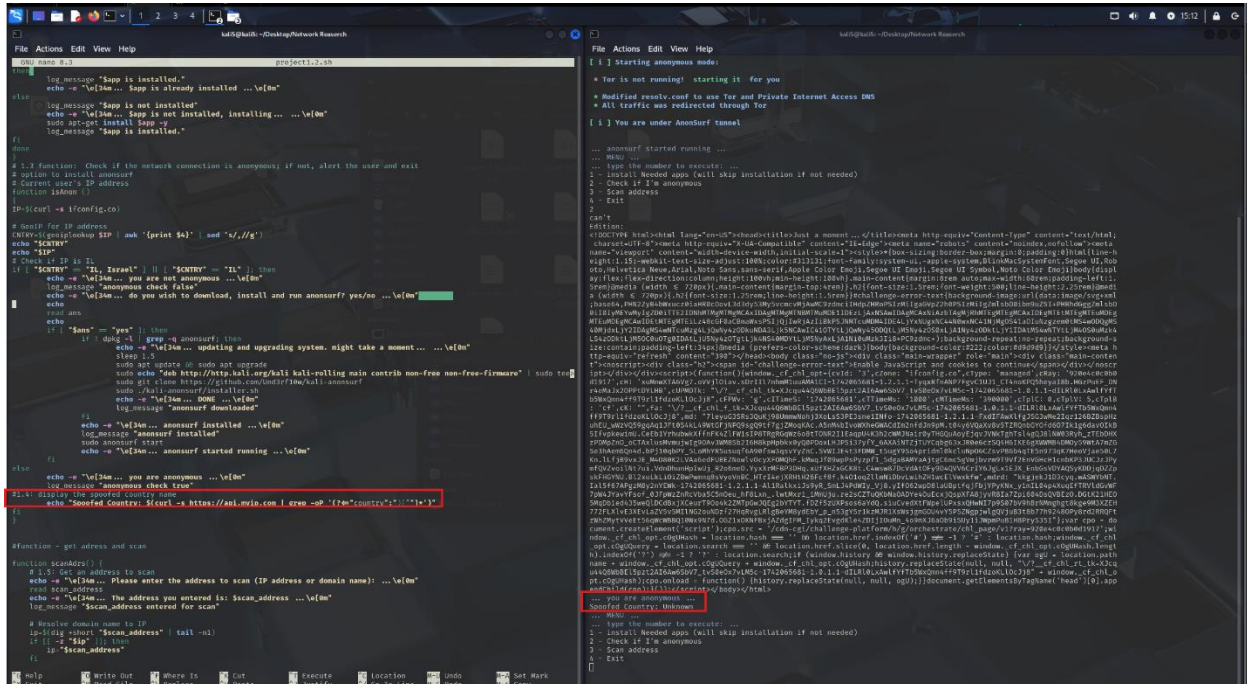
The "Comments" section includes the following text:

Use comments in your code to explain what you did.  
If you are using code from the internet, add credit and links.  
In the script, write the student's name and code, the class code, and the lecturer's name.









1. Installations and Anonymity Check
  - 1.1 Install the needed applications.
  - 1.2 If the applications are already installed, don't install them again.
  - 1.3 Check if the network connection is anonymous; if not, alert the user and exit.
  - 1.4 If the network connection is anonymous, display the spoofed country name.
  - 1.5 Allow the user to specify the address to scan via remote server; save into a variable.

- ### 3. Results
- 3.1 Save the Whois and Nmap data into files on the local computer.
  - 3.2 Create a log and audit your data collecting.

- #### 4. Creativity

## General

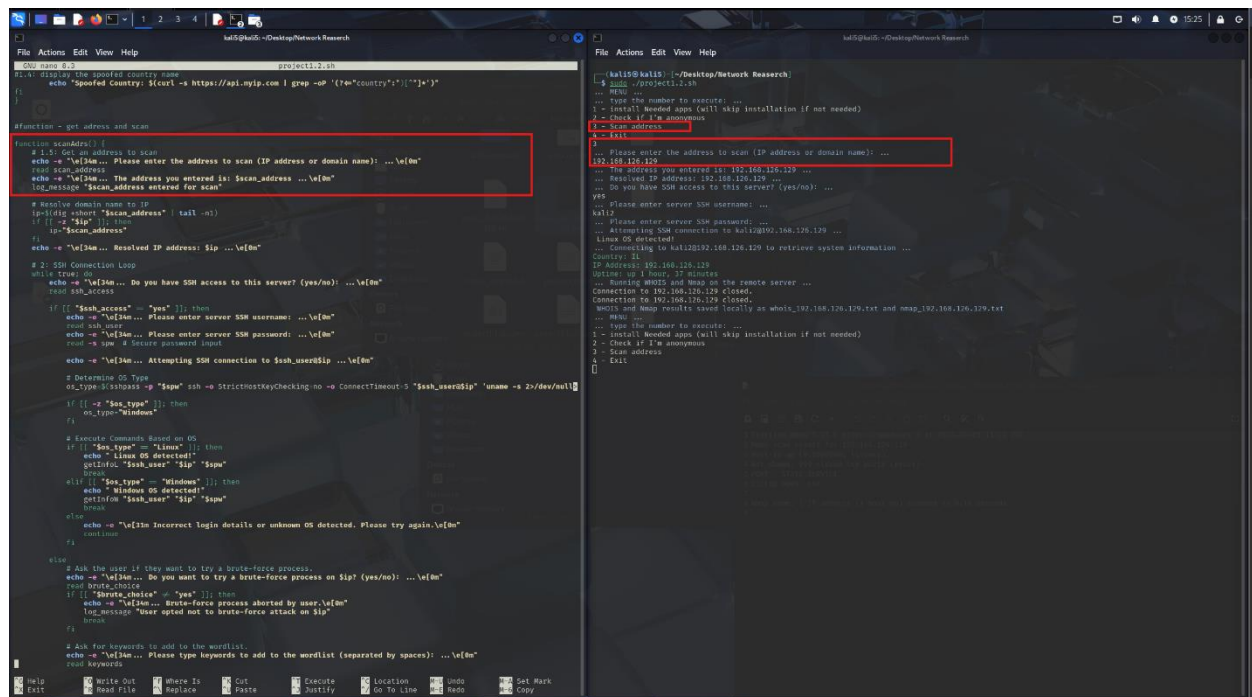
- Suggested tools: Sshpass, Nipe, Torify, Nmap, Whois.
- Everything other than the user input should be automated.
- Use functions.

### Comments

Use comments in your code to explain what you did.

If you are using code from the internet, add credit and links.

In the script, write the student's name and code, the class code, and the lecturer's name.



File Editor

Ctrl+S Save

Ctrl+Z Undo

Ctrl+Y Redo

Ctrl+F Find

Ctrl+G Go to line

Ctrl+H Replace

Ctrl+B Bookmarks

Ctrl+W Close

Ctrl+N New

Ctrl+O Open

Ctrl+S Save

Ctrl+P Print

Ctrl+Q Quit

94f50 ... 4.pdf

סיון תשפ"ו

סימולטור

האקטיביות והפרוייקטים

PROJECT: REMOTE CONTROL

## Project Structure

- Installations and Anonymity Check**
  - 1.1 Install the needed applications.
  - 1.2 If the applications are already installed, don't install them again.
  - 1.3 Check if the network connection is anonymous; if not, alert the user and exit.
  - 1.4 If the network connection is anonymous, display the spoofed country name.
  - 1.5 Allow the user to specify the address to scan via remote server; save into a variable.
- Automatically Connect and Execute Commands on the Remote Server via SSH**
  - 2.1 Display the details of the remote server (country, IP, and Uptime).
  - 2.2 Get the remote server to check the Whois of the given address.
  - 2.3 Get the remote server to scan for open ports on the given address.
- Results**
  - 3.1 Save the Whois and Nmap data into files on the local computer.
  - 3.2 Create a log and audit your data collecting.
- Creativity**

## General

- Suggested tools: Sshpass, Nmap, Torify, Nmap, Whois.
- Everything other than the user input should be automated.
- Use functions.

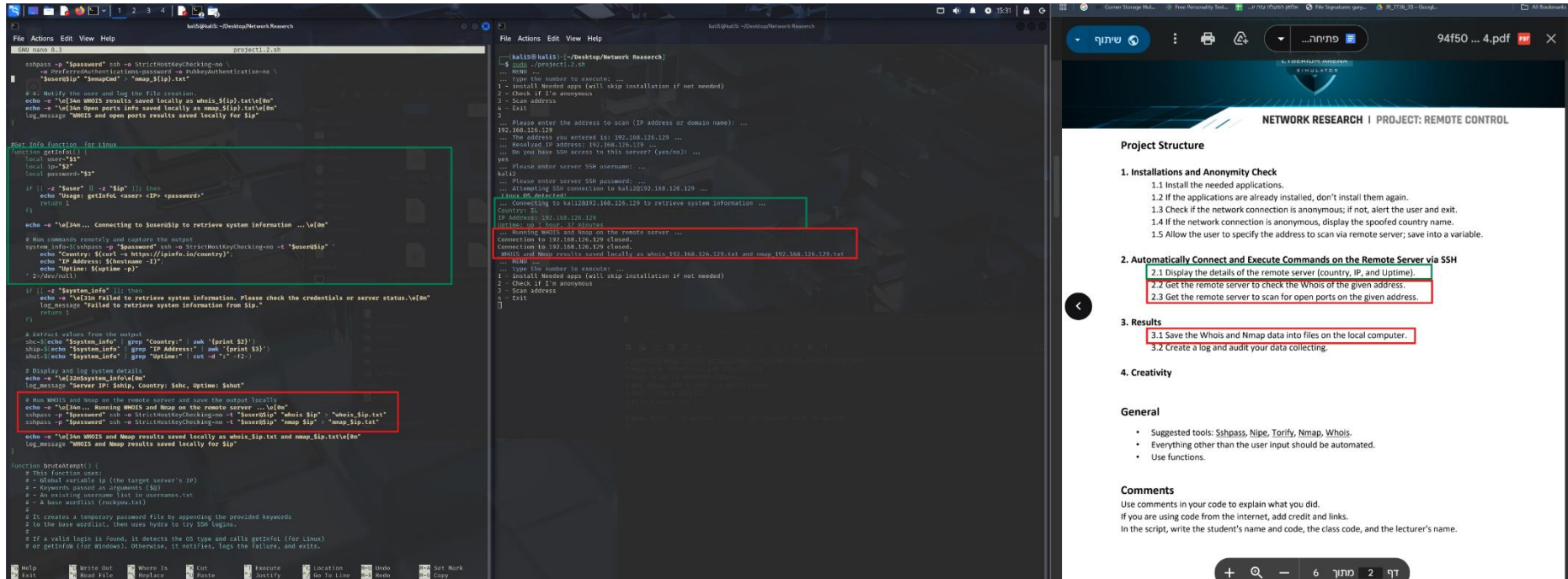
## Comments

Use comments in your code to explain what you did.  
If you are using code from the internet, add credit and links.  
In the script, write the student's name and code, the class code, and the lecturer's name.

+ 🔍 -

6 סיון תשפ"ו 2 דף

## 2. Automatically Connect and Execute Commands on the Remote Server via SSH



```
#!/usr/bin/perl

# Project 2.3.3
# This script is designed to connect to a remote server via SSH and execute commands.
# It uses the 'sshpass' utility to handle password prompts.
# The script is designed to be run on a Linux system.

# Get system information
function getinfo() {
    local user="root"
    local ip="192.168.1.100"
    local password="root"

    if [[ -z "$user" ]] || [[ -z "$ip" ]] || [[ -z "$password" ]]; then
        echo "Usage: getinfo <user> <ip> <password>"
        return 1
    fi

    echo "Connecting to $user@$ip to retrieve system information ..."
    sshpass -p "$password" ssh -o StrictHostKeyChecking=no -t "$user@$ip" 'cat /etc/passwd'
    echo "Running nmap and nmap results saved locally as nmap.$ip.txt"
    sshpass -p "$password" ssh -o StrictHostKeyChecking=no -t "$user@$ip" 'nmap -sP -sC -sV $ip'
    echo "nmap results saved locally as nmap.$ip.txt"
    log_message "nmap results saved locally as nmap.$ip.txt"
}

# Main function
main() {
    getinfo
}

main
```

Terminal Output:

```
kal15@kali:~/Desktop/Network Research$ ./project2.3.3
Connecting to root@192.168.1.100 to retrieve system information ...
Running nmap and nmap results saved locally as nmap.192.168.1.100.txt
nmap results saved locally as nmap.192.168.1.100.txt
```

Project Structure

1. Installations and Anonymity Check
  - 1.1 Install the needed applications.
  - 1.2 If the applications are already installed, don't install them again.
  - 1.3 Check if the network connection is anonymous; if not, alert the user and exit.
  - 1.4 If the network connection is anonymous, display the spoofed country name.
  - 1.5 Allow the user to specify the address to scan via remote server; save into a variable.
2. Automatically Connect and Execute Commands on the Remote Server via SSH
  - 2.1 Display the details of the remote server (country, IP, and Uptime).
  - 2.2 Get the remote server to check the Whois of the given address.
  - 2.3 Get the remote server to scan for open ports on the given address.
3. Results
  - 3.1 Save the Whois and Nmap data into files on the local computer.
  - 3.2 Create a log and audit your data collecting.
4. Creativity

General

- Suggested tools: `sshpass`, `Nmap`, `Torify`, `Nmap`, `Whois`.
- Everything other than the user input should be automated.
- Use functions.

Comments

Use comments in your code to explain what you did.  
If you are using code from the internet, add credit and links.  
In the script, write the student's name and code, the class code, and the lecturer's name.

### 3. Results

[illegible]

94f50 ... 4.pdf

... פתיחה

Free Password Tools... File Signatures guru... 7/7/18 - Google

סימולטור

NETWORK RESEARCH | PROJECT: REMOTE CONTROL

## Project Structure

- Installations and Anonymity Check**
  - 1.1 Install the needed applications.
  - 1.2 If the applications are already installed, don't install them again.
  - 1.3 Check if the network connection is anonymous; if not, alert the user and exit.
  - 1.4 If the network connection is anonymous, display the spoofed country name.
  - 1.5 Allow the user to specify the address to scan via remote server; save into a variable.
- Automatically Connect and Execute Commands on the Remote Server via SSH**
  - 2.1 Display the details of the remote server (country, IP, and Uptime).
  - 2.2 Get the remote server to check the Whois of the given address.
  - 2.3 Get the remote server to scan for open ports on the given address.
- Results**
  - 3.1 Save the Whois and Nmap data info to files on the local computer.
  - 3.2 Create a log and audit your data collecting.
- Creativity**

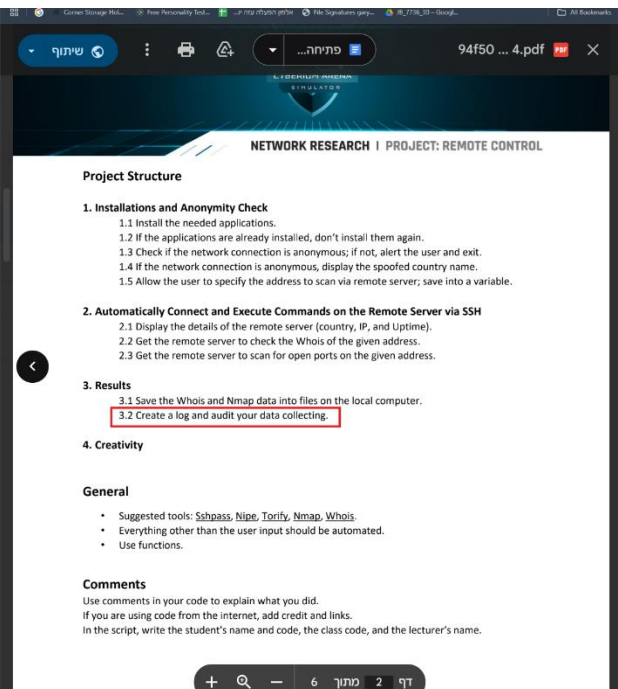
## General

- Suggested tools: Sshpass, Nlpe, Torify, Nmap, Whois.
- Everything other than the user input should be automated.
- Use functions.

## Comments

Use comments in your code to explain what you did.  
If you are using code from the internet, add credit and links.  
In the script, write the student's name and code, the class code, and the lecturer's name.





## 4. Creativity

- The script will detect if the given IP address runs Linux OS or Windows OS and will perform accordingly

The image displays a Kali Linux terminal window on the left and a PDF document on the right. The terminal window shows a script being executed, which prompts the user to enter an IP address and a password. The script then attempts to connect to the server via SSH. The PDF document, titled "NETWORK RESEARCH | PROJECT: REMOTE CONTROL", outlines the project structure and includes a section for "4. Creativity".

```
#!/bin/bash
# Project: Remote Control
# Author: [Your Name]
# Version: 1.0

# Function to get IP address
function getIP() {
    local user=$1
    local ip=$2
    local password=$3

    if [[ -z "$user" || -z "$ip" || -z "$password" ]]; then
        echo "Usage: getIP user ip password"
        return 1
    fi

    echo "Connecting to Windows server: $ip .../v0m"

    # 1. Retrieve and display basic system info from the remote Windows server.
    local sysCmd="powershell -NoProfile -Command \"
    Write-Host 'Country: ' $(Invoke-WebRequest -Uri 'https://ipinfo.io/country') \
    Write-Host 'IP Address: ' $(Get-NetIPAddress -AddressFamily IPv4 | Select-Object -ExpandProperty IPAddress) \
    Write-Host 'Uptime: ' $(New-Object -Type System.Diagnostics.Stopwatch -Start $(Get-Childinstance Win32_OperatingSystem).LastBootTime).ToString()\""
    sshpass -H $password ssh -o StrictHostKeyChecking=no \
        -o PreferredAuthentications=password -o PubkeyAuthentication=no \
        $user@$ip "$sysCmd"

    # 2. Perform WHOIS lookup and save output locally.
    # Note: In the class class, we double the apostrophe in "couldn't" for a correct literal.
    local whoisCmd="powershell -NoProfile -Command \"
    (Get-Command whois -ErrorAction SilentlyContinue) | \
    try { $result = whois $ip 2>&1; Write-Output $result } catch { Write-Output 'WHOIS lookup error: ' $_ } \
    else { Write-Output 'couldn't get WHOIS info, it might not be installed' }\""
    sshpass -H $password ssh -o StrictHostKeyChecking=no \
        -o PreferredAuthentications=password -o PubkeyAuthentication=no \
        $user@$ip "$whoisCmd" > "whois_$ip.txt"

    # 3. Check for NMAP. If found, use it; if not, use Get-NetTCPConnection as an alternative.
    local nmapCmd="powershell -NoProfile -Command \"
    (Get-Command nmap -ErrorAction SilentlyContinue) | \
    try { $result = nmap -p- -oX $ip 2>&1; if (not $result) { Write-Output 'NMAP scan failed or returned no results.' } } \
    else { \
        Write-Output 'NMAP command not found on remote Windows server. Using Get-NetTCPConnection as alternative.'; \
        try { $result = Get-NetTCPConnection -State Listen | Format-Table -AutoSize | Out-String; Write-Output $result } catch { \
            Write-Output 'Get-NetTCPConnection failed.' \
        } \
    }\""
    sshpass -H $password ssh -o StrictHostKeyChecking=no \
        -o PreferredAuthentications=password -o PubkeyAuthentication=no \
        $user@$ip "$nmapCmd" > "nmap_$ip.txt"

    # 4. Notify the user and log the file creation.
    echo "WHOIS results saved locally as whois_$ip.txt"
    echo "NMAP Open ports info saved locally as nmap_$ip.txt"
    log_message "WHOIS and open ports results saved locally for $ip"
}
```

The PDF document, titled "NETWORK RESEARCH | PROJECT: REMOTE CONTROL", outlines the project structure and includes a section for "4. Creativity".

### Project Structure

1. Installations and Anonymity Check
  - 1.1 Install the needed applications.
  - 1.2 If the applications are already installed, don't install them again.
  - 1.3 Check if the network connection is anonymous; if not, alert the user and exit.
  - 1.4 If the network connection is anonymous, display the spoofed country name.
  - 1.5 Allow the user to specify the address to scan via remote server; save into a variable.
2. Automatically Connect and Execute Commands on the Remote Server via SSH
  - 2.1 Display the details of the remote server (country, IP, and Uptime).
  - 2.2 Get the remote server to check the Whois of the given address.
  - 2.3 Get the remote server to scan for open ports on the given address.
3. Results
  - 3.1 Save the Whois and Nmap data into files on the local computer.
  - 3.2 Create a log and audit your data collecting.
4. Creativity

### General

- Suggested tools: Sshpass, Nmap, Torify, Nmap, Whois.
- Everything other than the user input should be automated.
- Use functions.

### Comments

Use comments in your code to explain what you did.  
If you are using code from the internet, add credit and links.  
In the script, write the student's name and code, the class code, and the lecturer's name.

- In case The user doesn't have the username or password for the SSH connection, the script will offer a brute force option. It will also ask the user for keywords he might think be relevant, and add them to the default rockyou file

