Project: Network remote control

# Table of contents

Introduction 3

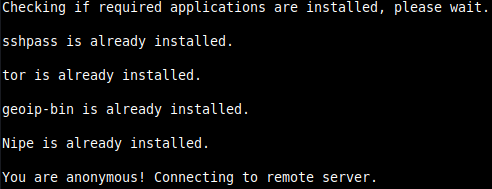
Methodologies 4

Discussion 5 – 11

References 12

**Introduction**

This project is about conducting various network-related tasks via automation using a bash script. With the help of text manipulations, the user can retrieve information like the IP address and the location of the server they are connected to. Not only that, the user can get the host machine to download and install the required tools to perform the information retrieval, all from the comfort of running the script and watching the machine equip itself and perform the tasks. The purpose of the script is to be able to plug and play on whichever host machine the user is at, producing the same desired results. As some of the tasks might be mundane and exhausting for the user to execute one by one, this is where scripting comes in. This report will cover the methods used to achieve the desired results, and what they look like in the script. New tools like sshpass will be explored to perform tasks without further user inputs. The report will also cover the transcripts of an FTP session, breaking down the events, and a brief explanation on how it works. An example of the script at work:



**Methodologies**

The first command used is dpkg. This will be used to check if a package is installed on the machine.



For the script to be fully automated, the -y flag is used with sudo apt-get install to automatically say yes during the installation prompt.



As nipe is not a package, an alternate way to check for its existence is to find its folder.



One way to hide the output, for example the unpacking/installation process, so as not to fill the screen with too much information, is to send them into null.



geoiplookup is used to find the country that the IP address is in.



To use ssh without needing user input(password), sshpass will be utilized.



For users to know how long has the system been running, uptime will be utilized.

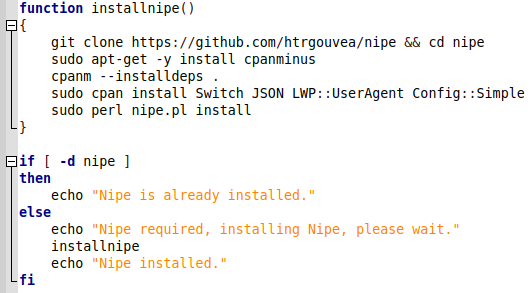


For automation, we do not want ssh to prompt us for an input. This will ask ssh to automatically add the host key.

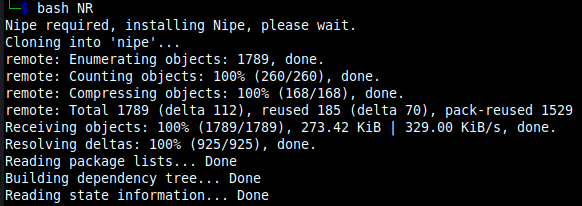


**Discussion**

First up, a function is stored for the installation steps for nipe if it isn’t present in the system. As nipe is not a package, the way to check for its directory’s existence is by using -d.



Example of the script running if nipe is not present.



The previous result contains the messages/information during the installation process which is not needed for the user to see. With > /dev/null 2>&1 which sends both output and error into the null, the result is cleaner as seen.

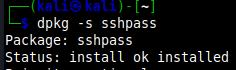


Using an array, the required applications can be stored under a single variable. Using in conjunction with a for loop, the applications in the array act as the range for the loop to run through.

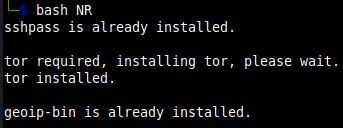


The -s flag for dpkg shows the status of the package. Again, sending the results to null as there will be overwhelming information. Using an if else statement, the script grep for the status either returning true or false which will determine if the application is installed on the system.

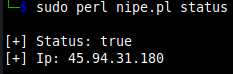




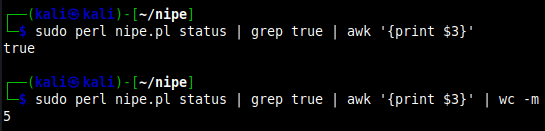
Example of the script running.

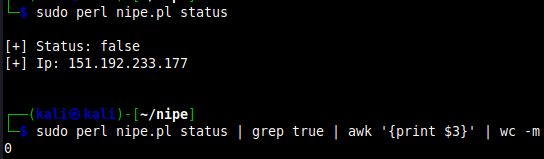


Next, check anonymity.

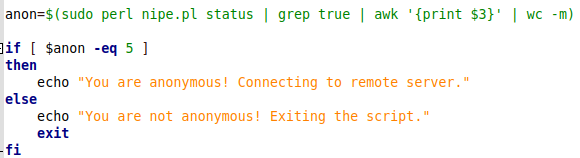


Status: true indicates that the anonymous connection is successful.

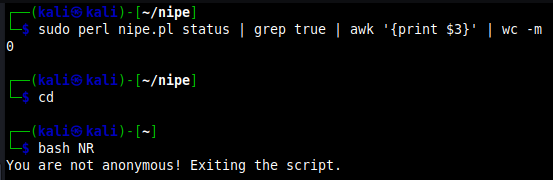




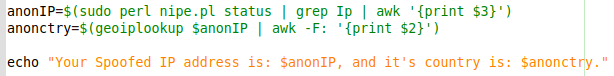
With true having a character count of 5, the if statement will be used to check if the word count matches, which will determine if the connection is anonymous.

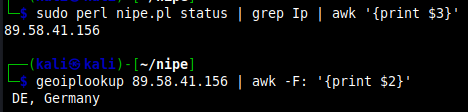


Example of the script running.

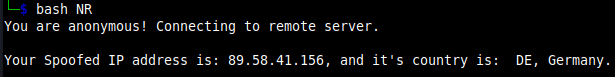


Once anonymous, the script will retrieve the spoofed IP address and country.





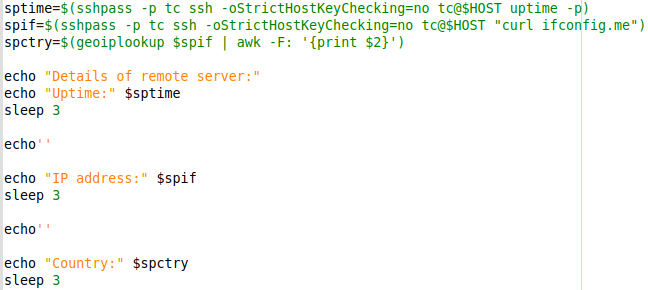
Example of the script running.



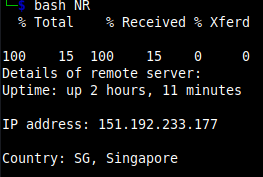
Variables are first stored to execute the respective commands.

Using sshpass, the user doesn’t have to input a password, maintaining the autonomous nature of the script. The -p flag is to store the password, which in this case is ‘tc’. HOST can be changed to match the IP of the respective server the user is accessing. tc@HOST refers to the user@IP\_Address, the remote server. The -p flag for uptime will display the output in human-readable format.



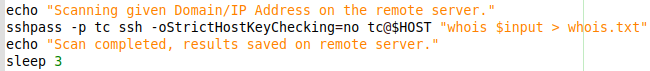


Example of the script running.

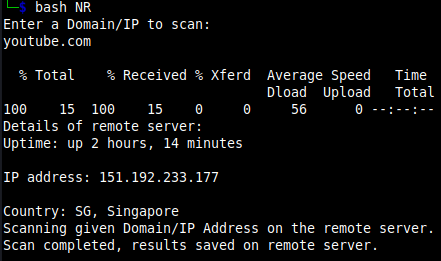


After sshpass, perform whois on the previously given input, and save the results into whois.txt.



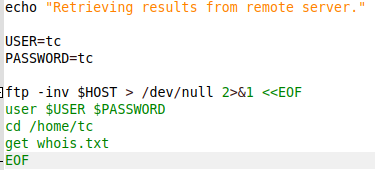


Example of the script running with domain/IP address scanned and saved onto the remote server.

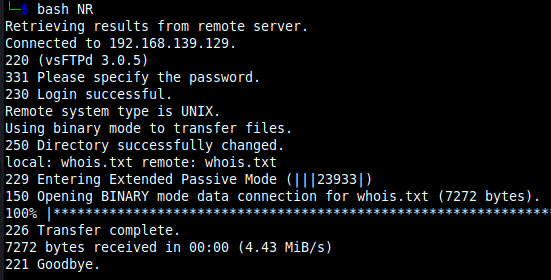




Next, the script will utilize ftp to retrieve the previously saved result from the remote server. In this part, the script will use ‘heredocs’. This is to enable entering multiple lines of input into an interactive prompt, as seen from the <<EOF and ending with EOF. Password and username are stored as variables so that there won’t be any interaction from the user. The -i flag turns off prompting if there are multiple file transfers. The -n flag disables autologin. The -v flag shows all responses from the remote server, as well as reports on data transfer statistics.



Example of the script running.



**References**

ioflood.com. (n.d.). *Bash Script Array Looping | How to Iterate Through Values*. [online] Available at: https://ioflood.com/blog/bash-loop-through-array/#:~:text=When%20it%20comes%20to%20arrays [Accessed 21 Mar. 2024].

www.baeldung.com. (2023). *How to Check Whether a Package Is Installed Through a Package Manager | Baeldung on Linux*. [online] Available at: https://www.baeldung.com/linux/check-how-package-installed [Accessed 21 Mar. 2024].

Stack Overflow. (n.d.). *Change the current directory from a Bash script*. [online] Available at: https://stackoverflow.com/questions/874452/change-the-current-directory-from-a-bash-script [Accessed 21 Mar. 2024].

Delony, D. (2022). *How to Geolocate IP Addresses on Linux Using geoiplookup*. [online] MUO. Available at: https://www.makeuseof.com/geolocate-ip-addresses-on-linux/ [Accessed 21 Mar. 2024].

Krishnakumar (2015). *How to Automate FTP transfers in Linux Shell Scripting*. [online] Eduonix Blog. Available at: https://blog.eduonix.com/2015/12/how-to-automate-ftp-transfers-in-linux-shell-scripting/ [Accessed 21 Mar. 2024].

Gerardi, R. (2020). *An introduction to using tcpdump at the Linux command line*. [online] Opensource.com. Available at: <https://opensource.com/article/18/10/introduction-tcpdump>.

Super User. (n.d.). *What are the restrictions to ssh StrictHostKeyChecking=no?* [online] Available at: https://superuser.com/questions/1751932/what-are-the-restrictions-to-ssh-stricthostkeychecking-no [Accessed 21 Mar. 2024].

December 2, A.V.G.L. updated: and Comments, 2020 0 (2019). *Check if a directory exists in Linux or Unix shell*. [online] nixCraft. Available at: <https://www.cyberciti.biz/faq/check-if-a-directory-exists-in-linux-or-unix-shell/>.

deepika92 (2021). *File Transfer Protocol (FTP)*. [online] GeeksforGeeks. Available at: <https://www.geeksforgeeks.org/file-transfer-protocol-ftp/>.

Wikipedia. (2021). *List of FTP server return codes*. [online] Available at: <https://en.wikipedia.org/wiki/List_of_FTP_server_return_codes>.

datatracker.ietf.org. (n.d.). *rfc959*. [online] Available at: <https://datatracker.ietf.org/doc/html/rfc959>.

Wikipedia. (2021). *List of FTP commands*. [online] Available at: https://en.wikipedia.org/wiki/List\_of\_FTP\_commands.