Lab *06– Scripting*

Name: Logan Weaver

Course/Section: IS-1003-01T

Date: 11/19/2024

INTRODUCTION

In this lab I learned how to write a bash script to display the current date, time and how to use nmap and curl to display my ip address and how to use a caesar cipher.

BREAKPOINT 1

Firstly I created a new file location called lab-06, I made it read and writable with chmod, and created a new bash script using the random number generator. I believe that writing a bash script holds many benefits over just writing these commands in the terminal. These include being able to replicate your commands by just calling the script, being able to see and understand your code better and it is way faster.

```
logan-weaver@loganweaver-VirtualBox:~$ cd lab-06
bash: cd: lab-06: No such file or directory
logan-weaver@loganweaver-VirtualBox:~$ cd Documents
logan-weaver@loganweaver-VirtualBox:~/Documents$ ls
lab-06  random-repo  unzip
logan-weaver@loganweaver-VirtualBox:~/Documents$ cd lab-06
logan-weaver@loganweaver-VirtualBox:~/Documents/lab-06$ echo $RANDOM
29200
logan-weaver@loganweaver-VirtualBox:~/Documents/lab-06$ touch weaver-29200.sh
logan-weaver@loganweaver-VirtualBox:~/Documents/lab-06$ ls -l
total 0
-rw-rw-r-- 1 logan-weaver logan-weaver 0 Nov 19 13:00 weaver-29200.sh
logan-weaver@loganweaver-VirtualBox:~/Documents/lab-06$ chmod +x weaver-29200.sh
logan-weaver@loganweaver-VirtualBox:~/Documents/lab-06$ ls -l
total 0
-rwxrwxr-x 1 logan-weaver logan-weaver 0 Nov 19 13:00 weaver-29200.sh
logan-weaver@loganweaver-VirtualBox:~/Documents/lab-06$ nano weaver-29200.sh
logan-weaver@loganweaver-VirtualBox:~/Documents/lab-06$ ./weaver-29200.sh
logan-weaver
Tue Nov 19 01:02:06 PM CST 2024
logan-weaver@loganweaver-VirtualBox:~/Documents/lab-06$
```

```
logan-weaver@loganweaver-VirtualBox:~/Documents/lab-06$ ./weaver-29200.sh
logan-weaver
Tue Nov 19 01:02:06 PM CST 2024
```

BREAKPOINT 2

In this breakpoint I fleshed out the bash script to display public, private ip addresses and open ports. I also created a cipher function. I included many comments and breaks in the code for readability because if code cannot be easily parsed, then it will be a lot harder to debug.

```
Logan-weaver@loganweaver-VirtualBox:~/Documents/lab-06$ ./weaver-29200.sh
logan-weaver
Tue Nov 19 01:19:43 PM CST 2024
todays date:
Private IP address:
Public IP adress:
open ports:
The cipher:
```

This screenshot shows the output of the bash script in my terminal.

```bash
#!/bin/bash

whoami
date

#############################################################FUNCTION BODIES########################
# A function to display the date and time

current_date() {
  # Display the current date and time
  echo "todays date: "
}

# A function to display the IP adresses and ports on this machine
ip_ports() {
  # Display prIvate IP address using hostname
  echo "Private IP address: "
  #Display public IP adress using ifconfig
  echo "Public IP adress: "
  #Display open ports using nmap
  echo "open ports: "
}


# A function to encrypt and decrypt a string using a sumple Caesar cipher
cipher () {

echo "The cipher: "

}




#############################################################FUNCTION CLASS########################
#displays current date
current_date
#displays open ip  ports, public and private ip adresses
ip_ports
#encrypts and decrypts a string using a caesar cipher
cipher
```

In this screenshot I am showing the layout of my bash script shows my name and the current date and time, along with other functions to get ip addresses and a cipher. In this screenshot I am showing the layout of my bash script shows my name and the current date and time, along with other functions to get ip addresses and a cipher.

BREAKPOINT 3

In this breakpoint I programmed a few lines of code to display the current date and time and display them in the terminal when the script is called. Our bash scripts always start out with #!/bin/bash, and comment by using '#'.

```
Tue Nov 19 01:25:54 PM CST 2024

Today's date: 11-19-24
The current time: 13:25:54 PM

Private IP address:
Public IP adress:
```

This screenshot shows my terminal output showing today's date and time.

```bash
#!/bin/bash

whoami
date

###############################################################FUNCTION BODIES#######################
# A function to display the date and time

current_date() {
  # Display the current date
  today=$(date +"%m-%d-%y")
  echo -e "\nToday's date: $today "

  # Display the current time
  current_time=$(date +"%H:%M:%S %p")
  echo -e "The current time: $current_time\n"
}

# A function to display the IP adresses and ports on this machine
ip_ports() {
  # Display prIvate IP address using hostname
  echo "Private IP address: "
  #Display public IP adress using ifconfig
  echo "Public IP adress: "
  #Display open ports using nmap
  echo "open ports: "
}


# A function to encrypt and decrypt a string using a sumple Caesar cipher
cipher () {

echo "The cipher: "

}




###############################################################FUNCTION CLASS#######################
#displays current date
current_date
#displays open ip  ports, public and private ip adresses
ip_ports
#encrypts and decrypts a string using a caesar cipher
cipher
```

This screenshot shows my bash script and how I programmed it to display the date and time.

BREAKPOINT 4

In this breakpoint I used curl and nmap to find and display my public and private ip address, and displayed it in the terminal. From my research I have found that curl is a command-line tool used for transferring data over various network protocols, and nmap is used for finding open ports and is used to map network infrastructures.

```
whoami
date

############################################################FUNCTION BODIES#########################
# A function to display the date and time

current_date() {
  # Display the current date
  today=$(date +"%m-%d-%y")
  echo -e "\nToday's date: $today "

  # Display the current time
  current_time=$(date +"%H:%M:%S %p")
  echo -e "The current time: $current_time\n"
}

# A function to display the IP adresses and ports on this machine
ip_ports() {
  # Display prIvate IP address using hostname
  echo "Private IP address: "
  hostname -I

  #Display public IP adress using ifconfig
  echo "Public IP adress: "
  curl ifconfig.me
  #Display open ports using nmap
  echo "open ports: "
  nmap scanme.nmap.org
}
```

In this screenshot I used nmap and curl to find my ip address and open ports.

```
logan-weaver@loganweaver-VirtualBox:~/Documents/lab-06$ ./weaver-29200.sh
logan-weaver
Tue Nov 19 01:35:14 PM CST 2024

Today's date: 11-19-24
The current time: 13:35:14 PM

Private IP address:
10.0.2.15
Public IP adress:
67.11.200.141open ports:
Starting Nmap 7.80 ( https://nmap.org ) at 2024-11-19 13:35 CST
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.083s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 996 filtered ports
PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
9929/tcp  open  nping-echo
31337/tcp open  Elite

Nmap done: 1 IP address (1 host up) scanned in 7.12 seconds
```

In this screenshot I show what my script displays in the terminal .

BREAKPOINT 5

In this breakpoint I got to use a Caesar cipher to encrypt and decrypt a message. If I wanted to change the cipher from a three shift to a 5 I would simply change the current tr line to tr 'a-zA-Z' 'f-za-eF-ZA-E', which would change it accordingly.

```
whoami
date

###########################################################FUNCTION BODIES########################
# A function to display the date and time

current_date() {
  # Display the current date
  today=$(date +"%m-%d-%y")
  echo -e "\nToday's date: $today "

  # Display the current time
  current_time=$(date +"%H:%M:%S %p")
  echo -e "The current time: $current_time\n"
}

# A function to display the IP adresses and ports on this machine
ip_ports() {
  # Display prIvate IP address using hostname
  echo "Private IP address: "
  hostname -I

  #Display public IP adress using ifconfig
  echo "Public IP adress: "
  curl ifconfig.me
  #Display open ports using nmap
  echo "open ports: "
  nmap scanme.nmap.org
}


# A function to encrypt and decrypt a string using a sumple Caesar cipher
cipher () { run=true
while $run; do
  read -p "Let's encrypt a string! Enter a string of your choice (or '0' to quit): " input
if [[ "$input" == "0" ]]; then
  echo "Thank you! Bye!!"
  run=false
else
  encrypted=$(echo "$input" | tr 'a-zA-Z' 'd-za-cD-ZA-C') decrypted=$(echo "$encrypted" | tr 'a-zA-Z' 'x-za-wX-ZA-W')
  echo "Encrypted: $encrypted"
  echo "Decrypted: $decrypted"
fi done
}
```

The screenshot above is showing my full bash script which tells the user the current time, date, ip addresses, open ports and encrypts and decrypts a string given by the user.

```
logan-weaver@loganweaver-VirtualBox:~/Documents/lab-06$ ./weaver-29200.sh
logan-weaver
Tue Nov 19 01:35:14 PM CST 2024

Today's date: 11-19-24
The current time: 13:35:14 PM

Private IP address:
10.0.2.15
Public IP adress:
67.11.200.141open ports:
Starting Nmap 7.80 ( https://nmap.org ) at 2024-11-19 13:35 CST
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.083s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 996 filtered ports
PORT       STATE SERVICE
22/tcp     open  ssh
80/tcp     open  http
9929/tcp   open  nping-echo
31337/tcp open  Elite

Nmap done: 1 IP address (1 host up) scanned in 7.12 seconds
Let's encrypt a string! Enter a string of your choice (or '0' to quit): "i am very sleepy"
Encrypted: "l dp yhub vohhsb"
Decrypted: "i am very sleepy"
Let's encrypt a string! Enter a string of your choice (or '0' to quit): '0'
Encrypted: '0'
Decrypted: '0'
Let's encrypt a string! Enter a string of your choice (or '0' to quit): 0
Thank you! Bye!!
```

This is the output of my completed bash script! It works well and was very fun to create.

CONCLUSION

I did not have many challenges or issues, I didn't even have to debug my code like I had to the last lab. This was my first time using nmap and I can not wait to use it in the future.

REFERENCES

R. Mitra, "*Lab 06: Scripting*," The University of Texas at San Antonio (2024). Last accessed: *11/19/2024*.

COLLABORATION

I worked alone.