SSH: A Secured Connection

November 14th, 2019

CSUF; CPSC 311

Professor Peralta

Midterm Paper

Chukwudi Ikem 889294070

Outline

1 Abstract

The purpose of this document is to inform the reader about what the Secure Shell is. I will start by giving a brief summary of how Secure Shell came to be. This includes the various methods of computer communication that were available in years past. SSH uses keys as a way to authenticate one computer with another, this will allow for password-free logins and other conveniences. I will go over useful SSH Tunneling functions such as port forwarding and ways to securely transfer files. You may wish to bypass some network restriction, this can be done with any type of port forwarding. Local port forwarding is how you bind a single port to the client to tunnel into another port on the server that is unrestricted. Dynamic Port Forwarding involves creating a SOCKS proxy at a local port. If port 9001 the home for the proxy, then we could configure our client to point to the SOCKS proxy at that port. File Transfers may be achieved using SCP or SFTP, standing for Secure Copy Protocol and Secure File Transfer Protocol respectively. These are commands that utilize the SSH Tunnel to secure file transfers. The last paragraph will detail the basics of how to use SSH, it is a very brief overview. The conclusion illustrates the utility that such computer functionality may bring to your everyday life.

2 Introduction

Remote shell protocols such as RSH and RLogin were meant to be the protocols that would allow for remote control of one computer by another. The purpose involved creating a network of computers that could execute shell commands remotely. Remote shell protocols, along with Telnet protocols, were supposed to advance the principles of client-server communication. Although remote protocols were insecure and seemingly amateurish when compared to modern-day protocols, there is no doubt that these protocols were essential to the emergence of SSH. SSH made an effort not only to secure any connection made by two computers. Also, to provide a significant amount of functionality that would allow a user to be sheltered from potential data theft while giving that same user a further sense of technological autonomy. The Secure Shell (SSH) has proven to be an essential protocol when attempting to establish a secure connection between two machines. Such a link allows you to treat the computer you connect to as your own. SSH allows you to copy data into the new computer, delete files, create files, and even go as far as opening applications on the server. Secure Shell served as an improvement to the previous methods that attempted this sort of client-server connection.

3 History of Computer Communication

A connection from one computer another has seen many reforms since the modems used in the 60s. Telnet and remote shell protocols were not the first to attempts at computer communication. Ironically, the main objective of most machines in the late 20th century was to boast processing power and not to be communication devices. Later on, developments on personal computers allowed for data exchange in various ways. One method as mentioned before is the modem.

> "Modem is short for modulate-demodulate - that's where it got its name. Modems enable the digital form of matter that a computer uses to communicate by the analog form of transmission of old-style telephone systems" (Peter)

In layman's terms, a modem is used to transform information in such a way that it can be transmitted over a telephone line to the "other end,". A modem virtually connects computers by telephone lines. The modem allowed for terminals to connect to computers over telephone lines. Hence there was a means of establishing, albeit insecure, connections between computers using a telephone network.

3.1 Telnet

By 1969 came the development of Telnet. Telnet is a set of client-server protocols which could allow a user of one computer to take control of another without the need of them being on the same local network. Yet, when Telnet developed, it was mostly used by large institutions that were generally around the same area. Although Telnet was sending non encrypted data across networks, it did not matter since those networks were connected in the same local environment. Yes, anyone with access to the Telnet connection could intercept packets of data and obtain valuable information. However, during that time, it would only be possible if you were in the vicinity of packet transfer.

3.2 The Security Concern

The concern for network security arose upon the advancements of domestic modems and new broadband connections. Transmission Control Protocol and Internet Protocol define how data exchanges between networks. TCP identifies how information is broken down into packets and distributed over the Internet. IP specifies the route each packet takes to arrive at the correct destination. As TCP/IP protocols were becoming the standard, communication between computers through wide area networks became more prevalent. As the commercial Internet began to grow in the early 1990s, so did the rise of internet hackers. It was no longer sufficient to create connections between computers. It became increasingly important to establish secure connections. Secure Shell sought to address many of the issues that occurred when establishing a connection between client and server (the server is the computer that you wish to control).

> "Secure Shell sought to provide authentication, encryption, and data integrity to secure network communications. Implementations of Secure Shell offer the following capabilities: a secure command-shell, secure file transfer, and remote access to a variety of TCP/IP applications via a secure tunnel" (VanDyke)

Sensitive data in our generation is prevalent not only in large institutions but small businesses and domestic households. With the release of the Secure Shell, setting up an SSH server and creating an encrypted connection is feasible.

4 SSH Keys

As stated previously, SSH is a protocol that works in a client-server model. This model implies that the connection is established by the SSH client connecting to the SSH server. In this particular model, not only are the packets of data traveling between the client and server encrypted. Algorithms are encrypting the authentication between the client and server itself. The "setup" phase handled by the client and server follows a particular pattern. The authentication process requires that the server be able to receive a public key, a sort of identification protocol generated by the client. Once the public key has been generated, the client passes it over to the server. The server checks to see if it has authorized you with that key previously. It's mainly checking your identification to see if a connection (signature) has established already. For private keys, if you are authorized, the server will take note of that and send back a completely new key back to the client. The encrypted key that was sent back to the client can only open with the public key that was verified by the server. You may now use that key to gain access to your data or send it back to the server. This method of public and private key verification involves a plethora of redundancies to ensure safe connections, even during authentication. There are different algorithms Secure Shell will utilize (asymmetric or symmetric encryption) to handle the flow of data between client and server. Various hashing algorithms are implemented to change according to the size and type of data flowing in the connection. These precautions secure the authentication and integrity of the data sent between the two computers.

5 SSH Tunneling

Not only does SSH serve as a tool for protecting your data, it is often a way to create an encrypted tunnel over a network connection. This tunnel may allow you to transfer data, bypass firewalls, or even certain internet filters. SSH generates its appeal by offering an immense amount of functionality, one of them being the idea of port forwarding. A port in computer networking is an endpoint of communication. It is often used to represent a particular protocol or process. Similar to an extension number schools may use to contact other professors and administrators at the school.

> Computers also have a sort of main address and a set of extension numbers (port numbers) that can handle incoming and outgoing connections (Mitchell). For example. Port 80 is a port number that is commonly assigned to represent internet communication processes (Kumar).

The idea of SSH Tunneling is that we can "forward" a port over the SSH Tunnel. A port forward means if a port was blocked initially for whatever reason – we could bypass the restriction by forwarding the port over to a computer that does not have that port blocked.

5.1 Local Port Forwarding

The idea behind local port forwarding is that you can bind some port on the client computer to listen for web requests made on the server computer. Assuming the SSH tunnel was created, the client computer no longer has to worry about a potential firewall that may be blocking the connection. It is the job of the server to handle the requests bypassing the firewall. Assuming the server would not have any port restrictions, this should be a reasonably simple task. The port that is bound to the client is then used as a catalyst for handling requests made on the open server port. The blocked port on the client is the port that is open on the server, by using an SSH tunnel we can utilize a new port on the client to handle data requests made on the unblocked server port. An example from Buddhika Chamith's blog best illustrates the machinations of local port forwarding. Say that you are at work, and there is a website that happens to be blocked. If the SSH server on your home computer is already set up, then you can bypass the website restriction (Chamith). The idea being that you bind a free port ("port 9001 for example") on the work computer to listen for local requests made by the home computer. The open port is waiting for data packets to be sent from the home computer. The home computer can make the connection to the blocked website, assuming you do not have restrictions of your own. The work computer must only listen for requests made to that blocked website from the home computer. Once the connection is established, and the request is sent, you may now visit the blocked website on the client computer by accessing the bound port that connects to the server. A simple request to "http://localhost:9001" will double as a request to your home computer, which is connected to that "blocked" website. It is important to note that while the connection from work to home is encrypted, the connection between home and the blocked website is not.

5.1.1 Reverse Port Forwarding

Reverse Port Forwarding is the process of connecting a "home" computer to a "work" computer. This port forward reverses the operation of local port forwarding. As an example, we may use our home computer as the client to listen for requests made on the work computer instead of binding a free port on the work computer to listen for requests made on the home computer. You may wish to access an internal site belonging to the "work" server from home. Assuming that an SSH tunnel has been established along with the proper port binding. Visiting "http://localhost:9001" in your home web browser may allow you to access the work computer's internal site and relay that response back to the client.

5.2 Dynamic Port Forwarding

Local Port Forwarding requires that you specify some destination on the server and a port on the client during the SSH protocol. In dynamic forwarding, you do not have to specify a single port or a destination. Instead, you start the forward by creating a SOCKS proxy. A proxy is a particular type of server that operates in such a way that it can receive requests from clients to other servers. The proxy is the medium between the client and the server. When a SOCKS proxy is used, the data packets will flow through SOCKS and onto the requested IP address. The power of dynamic port forwarding comes from the ability to create a SOCKS proxy at a local port. In doing so, the proxy that you created can route TCP/IP connections directly to the web-page. If port 9001 is the home for the proxy, then we could configure our client to point to the SOCKS proxy at 9001 instead of making a direct request to a web page. With this feature, you may also bypass firewalls and network restrictions that you may have otherwise had in place.

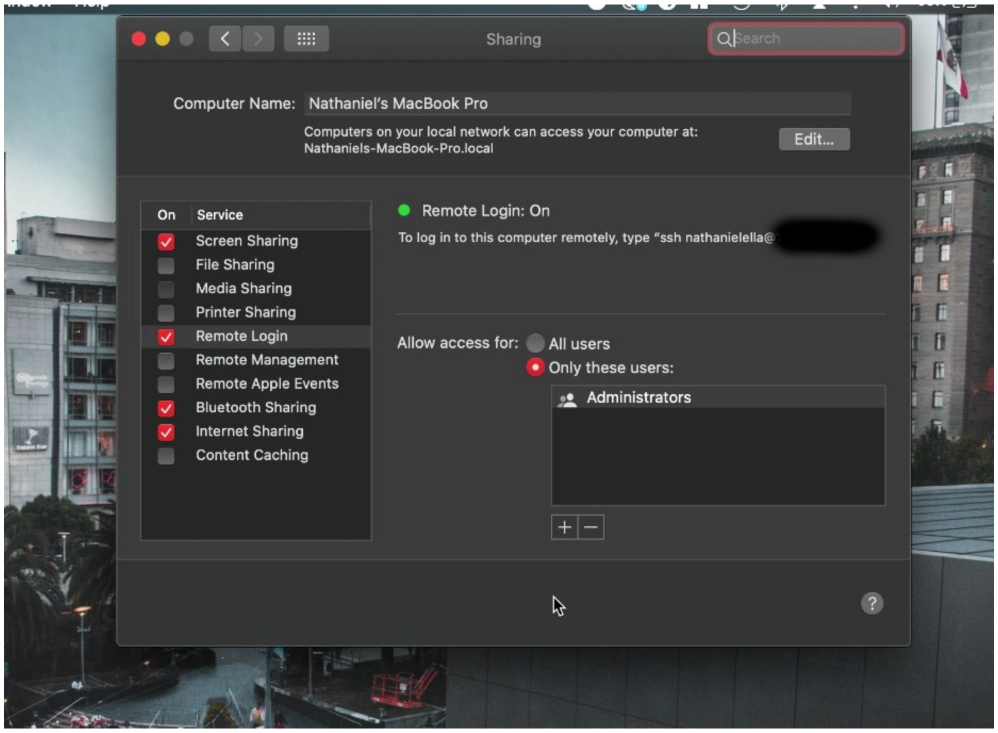
6 Secure Copy. Secure File Transfer Protocol

While port forwarding may allow you to bypass the inconveniences of network restrictions, it may also allow you to avoid the inconveniences of distance. You may wish to be able to securely send files over an SSH tunnel from one computer to another. The Secure Copy Protocol is one such protocol that permits safe transfers of files on a network using TCP Port 22. The same rules of authentication and encryption applies, by using the SSH keys and associated security protocols you can make quick transfers between client and server. By default, SCP excels in file transfer speed because that is all it does, it will not attempt to authenticate the file transferred. In some ways, SFTP is an improvement on SCP because of the functionality that comes with it. The Secure File Transfer Protocol is not built solely for file transfers. You can create and delete directories and files as you could on any standard command-line interface using specific commands. Since both SCP and SFTP are based on SSH, running over TCP port 22, they both run using the same general format as other SSH protocols and remote tools discussed previously. Secure Copy Protocol and Secure File Transfer Protocol illustrate the autonomy that SSH remoting provides. SSH allows for even greater control over computers and network. Not only are you able to securely elude network restrictions with port forwarding, you may secure file transfers from client to server using similar principles.

7 How to Use SSH

SSH provides many tools while demonstrating the power of SSH. Putting it into practice will provide true evidence for how truly extraordinary it is. For UNIX based operating systems such as macOS and Linux, SSH client-server tools are already built-in. For users wishing to utilize these operations on Windows they must refer to an application called PuTTY[1]. In this example I will demonstrate the instructions for establishing a connection from a Linux computer (client) to Nathaniel's MacBook (server). Since the MacBook is the computer that I am trying to connect to, I need to figure out a way to pull up the SSH settings. Luckily enough for this particular model, it is in the Sharing section in the Settings.

1 "Home." SSH.COM, www.ssh.com/ssh/putty/windows/.



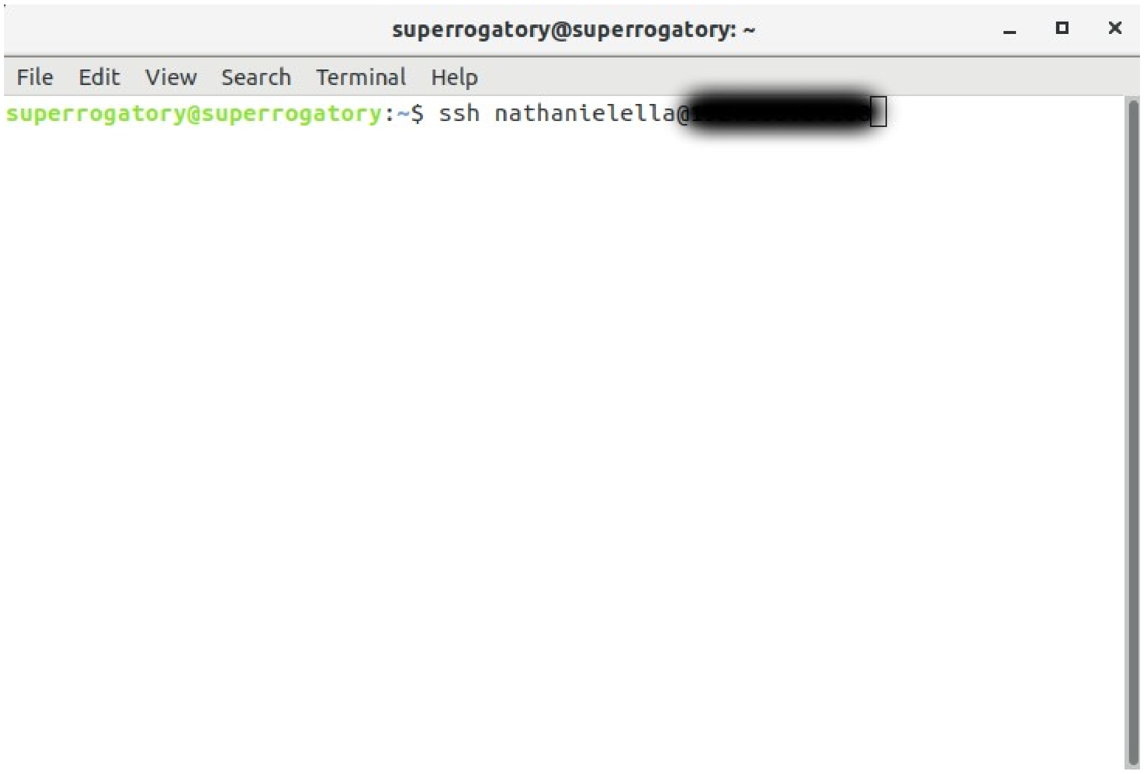
Computer Name: Nathaniel's MacBook Pro
Computers on your local network can access your computer at:
Nathaniels-MacBook-Pro.local

On	Service
✓	Screen Sharing
	File Sharing
	Media Sharing
	Printer Sharing
✓	Remote Login
	Remote Management
	Remote Apple Events
✓	Bluetooth Sharing
✓	Internet Sharing
	Content Caching

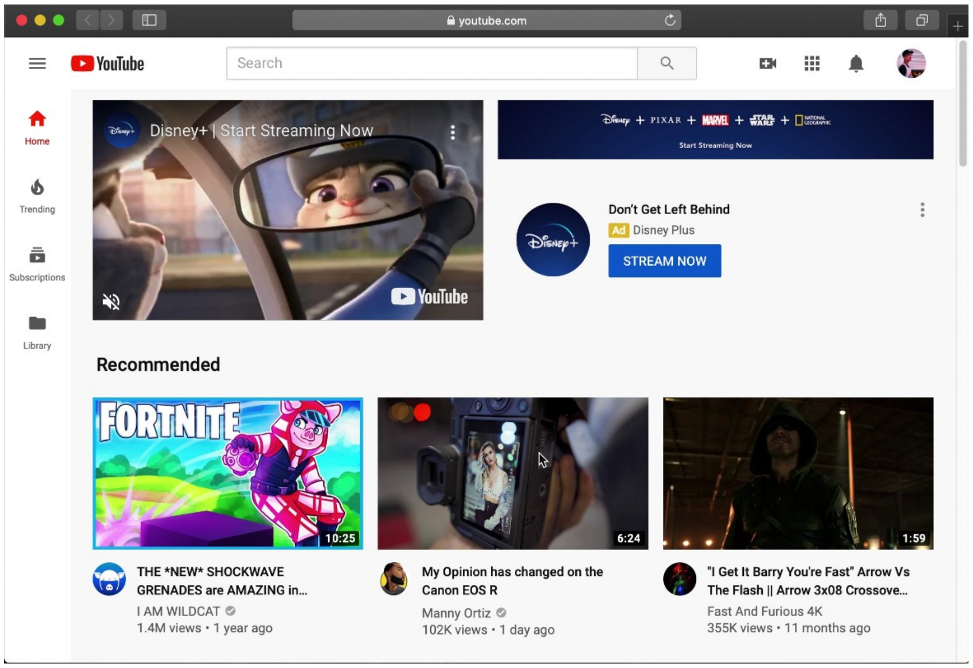
Remote Login: On
To log in to this computer remotely, type "ssh nathanielella@[redacted]

Allow access for: All users [✓ Only these users:]
Administrators

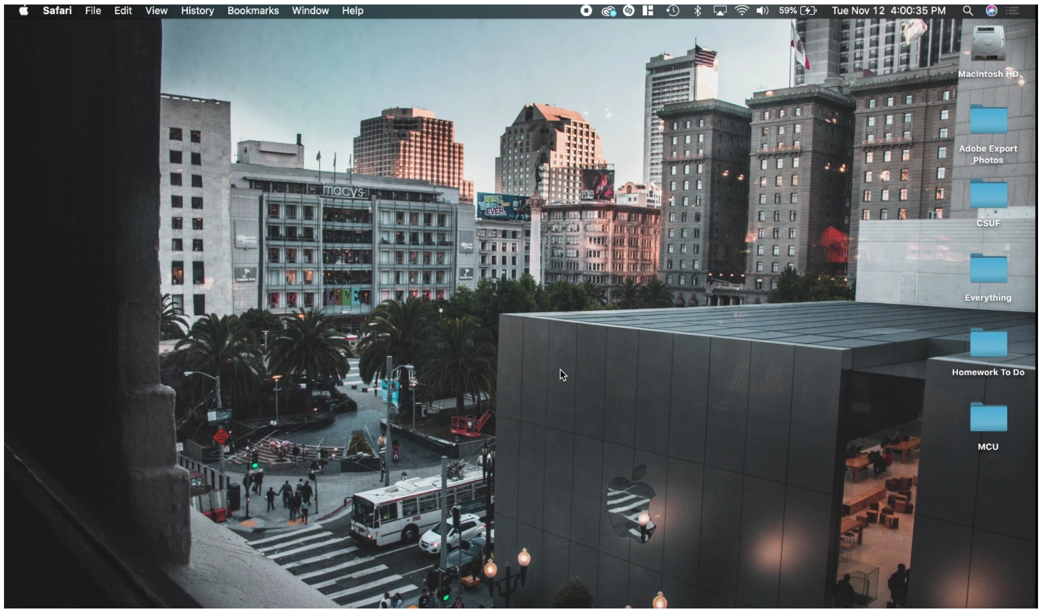
If remote login is turned on, then there will be instructions on what to type in on the client computer. On my personal terminal, I will type in ssh nathanielella@ipaddress, and this should prompt for Nathaniel Ella's personal password that he uses to login to his computer.

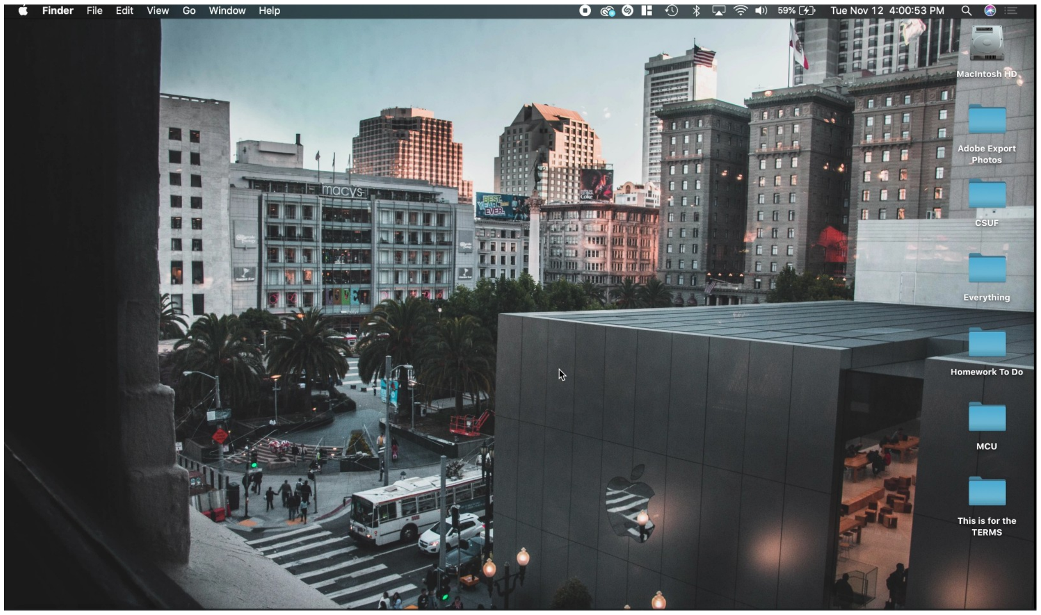

superrogatory@superrogatory: ~
File Edit View Search Terminal Help
```
superrogatory@superrogatory:~$ ssh nathanielella@[redacted]
```

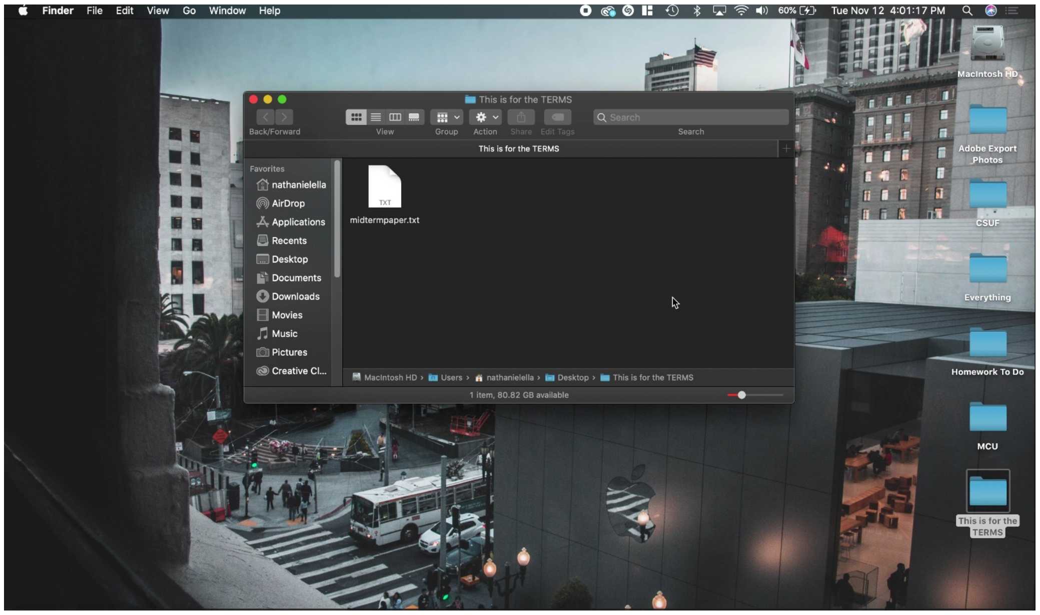
Immediately after the password had been entered, then you will notice the terminal name that is usually yours turn into the name of the computer that was securely connected to. At this point, you can use standard terminal commands to modify text files, create and delete files, open applications, and more. For example, "open -a Safari.app "http://www.youtube.com" will open Safari on the MacBook, onto the YouTube web-page.


youtube.com
YouTube
Search
Home
Trending
Subscriptions
Library
Disney+ | Start Streaming Now
Don't Get Left Behind
Ad Disney Plus
STREAM NOW
Recommended
THE *NEW* SHOCKWAVE GRENADES are AMAZING in...
I AM WILDCAT
1.4M views • 1 year ago
My Opinion has changed on the Canon EOS R
Manny Ortiz
102K views • 1 day ago
"I Get It Barry You're Fast" Arrow Vs The Flash || Arrow 3x08 Crossove...
Fast And Furious 4K
355K views • 11 months ago

I may wish to make a new folder in the Desktop. This can be accomplished using the "mkdir [name]" command. Here I will create a new folder named "This if for the TERMS" and will create a new text file in there using the "touch [name].txt" command.


Safari File Edit View History Bookmarks Window Help
59% Tue Nov 12 4:00:35 PM
Macintosh HD
Adobe Export Photos
CSUF
Everything
Homework To Do
MCU





I may also wish to use a terminal text editor such as nano or vim to modify the text files that were just created. There is a great power in the ability to manipulate a computer from a remote location, and these commands only show half of the picture. We are still able to utilize different port forwarding methods, secure transfer protocols, and anything using the SSH Tunnel.

8 Conclusion

Remote shell and Telnet protocols provided a way for computers to communicate. Although the data sent through these older protocols were transferred in plain-text, a very insecure format, it was the principles behind such a new form of communication that allowed for an improved version of it to arise. Secure Shell not only provides encryption between the client and the server. SSH also ensures an authentication process that avoids any middle man attack. With an established secure tunnel between two computers, they can reap the benefit of port forwarding and secure file transfer protocols. SSH has proven to be a vital upgrade to the previous methods of transmitting data from one computer to another, and it provides the user with the ability to control their machines with security and precision.

Work Cited

Team, Cerberus. "SCP vs SFTP - Which Is Better for Your Needs?" Cerberus FTP Server, Cerberus, 20 Dec. 2018, www.cerberusftp.com/comparing-scp-vs-sftp-which-is-better/.

Knafo, Jenny. "How to Setup Dynamic Port Forwarding in Remote Desktop Manager." The Devolutions Blog, Devolutions, 18 Oct. 2018, blog.devolutions.net/2018/10/how-to-setup-dynamic-port-forwarding-in-remote-desktop-manager.

Mitchell, Bradley. "Port Numbers Used for TCP/IP Computer Networks." Lifewire, Lifewire, 5 Nov. 2019, www.lifewire.com/port-numbers-on-computer-networks-817939.

Kumar, Ranjit. "What Is Port 80? - Definition from Techopedia." Techopedia.com, 2015, www.techopedia.com/definition/15709/port-80.

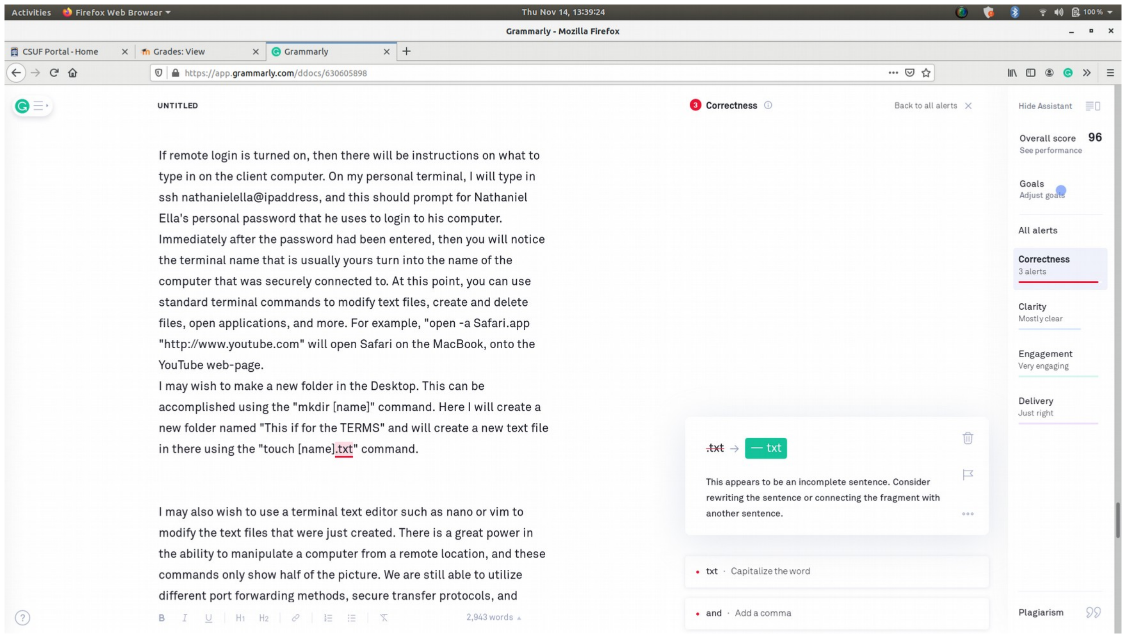
Chamith, Buddhika. "SSH Tunneling Explained." Source Open, 11 Nov. 2012, chamibuddhika.wordpress.com/2012/03/21/ssh-tunnelling-explained/.

Peter, Ian. "The History of Computers, Networks and Modems." Computers, Networks, and Modems, 2004, www.nethistory.info/History%20of%20the%20Internet/netsnmods.html.

"An Overview of the Secure Shell (SSH)." VanDyke, White Paper, 2008, www.vandyke.com/solutions/ssh_overview/ssh_overview.pdf.

Grammarly

The Grammarly red lines are for a text file extension name, I cannot change this. It also wanted to me to reform a work cited quotation. I also have video proof of me making all of the corrections. It seems like the same issues with Grammarly occur. One hour there are no errors, and another hour there are three errors.



If remote login is turned on, then there will be instructions on what to type in on the client computer. On my personal terminal, I will type in ssh nathanielella@ipaddress, and this should prompt for Nathaniel Ella's personal password that he uses to login to his computer. Immediately after the password had been entered, then you will notice the terminal name that is usually yours turn into the name of the computer that was securely connected to. At this point, you can use standard terminal commands to modify text files, create and delete files, open applications, and more. For example, "open -a Safari.app "http://www.youtube.com" will open Safari on the MacBook, onto the YouTube web-page.
I may wish to make a new folder in the Desktop. This can be accomplished using the "mkdir [name]" command. Here I will create a new folder named "This if for the TERMS" and will create a new text file in there using the "touch [name].txt" command.

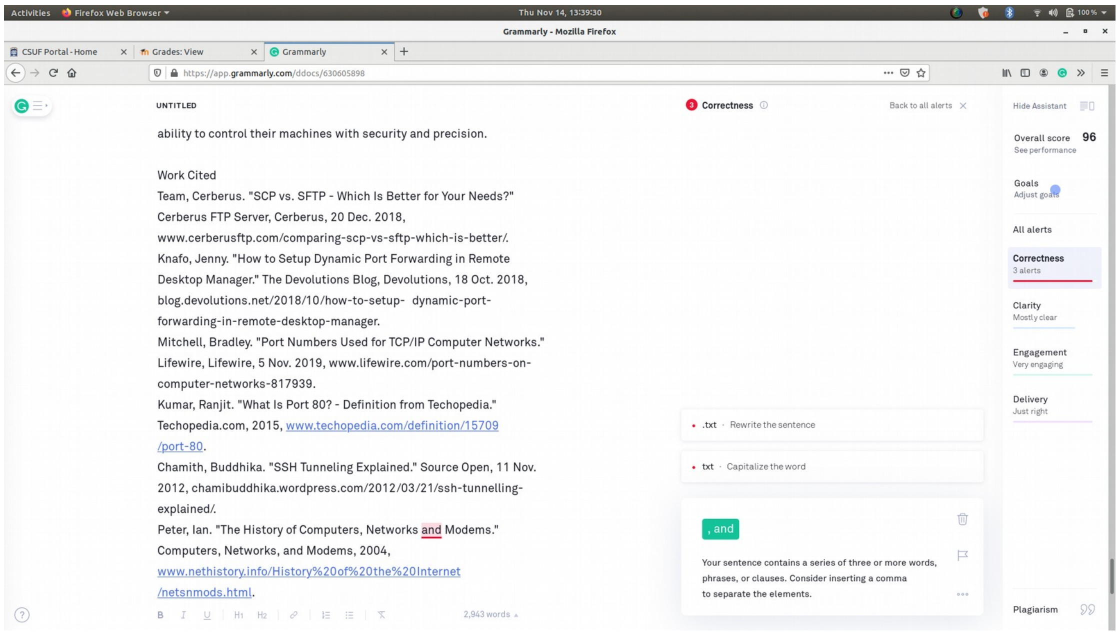
I may also wish to use a terminal text editor such as nano or vim to modify the text files that were just created. There is a great power in the ability to manipulate a computer from a remote location, and these commands only show half of the picture. We are still able to utilize different port forwarding methods, secure transfer protocols, and

.txt → — txt

This appears to be an incomplete sentence. Consider rewriting the sentence or connecting the fragment with another sentence.

txt · Capitalize the word

and · Add a comma



ability to control their machines with security and precision.

Work Cited
Team, Cerberus. "SCP vs. SFTP - Which Is Better for Your Needs?" Cerberus FTP Server, Cerberus, 20 Dec. 2018, www.cerberusftp.com/comparing-scp-vs-sftp-which-is-better/.
Knafo, Jenny. "How to Setup Dynamic Port Forwarding in Remote Desktop Manager." The Devolutions Blog, Devolutions, 18 Oct. 2018, blog.devolutions.net/2018/10/how-to-setup- dynamic-port-forwarding-in-remote-desktop-manager.
Mitchell, Bradley. "Port Numbers Used for TCP/IP Computer Networks." Lifewire, Lifewire, 5 Nov. 2019, www.lifewire.com/port-numbers-on-computer-networks-817939.
Kumar, Ranjit. "What Is Port 80? - Definition from Techopedia." Techopedia.com, 2015, www.techopedia.com/definition/15709/port-80.
Chamith, Buddhika. "SSH Tunneling Explained." Source Open, 11 Nov. 2012, chamibuddhika.wordpress.com/2012/03/21/ssh-tunnelling-explained/.
Peter, Ian. "The History of Computers, Networks and Modems." Computers, Networks, and Modems, 2004, www.nethistory.info/History%20of%20the%20Internet/netsnmods.html.

.txt · Rewrite the sentence

txt · Capitalize the word

, and

Your sentence contains a series of three or more words, phrases, or clauses. Consider inserting a comma to separate the elements.