**P2P File Transfer**

**Group Project**

Jia Da Wu, Keshawn Gosiengfiao, Lilia Barajas Lopez, Victor Llano Marotta

**Executive Summary**

According to MarketsandMarkets research study, “The demand for encryption software is likely to be driven by various factors, such as proliferation in the number of cyber-attacks…”. Taking this into consideration, how can we make sure that any of the information shared across a network is safe?

Our software company, P2P File Transfer, has created a safe way to share information by facilitating encrypted file sharing. Because of the rapid growth for the need of encryption use, our product offers tremendous advantages to company professionals, as well as the thousands of other people who use networks like the internet to share information every day.

Our software was developed in Python. It is built with email delivered download file options that consist of “Receiver” and “Sender”. P2P File Transfer software starts with a Registration PIN, continues with Two-Factor Authentication powered by Google Authenticator, and finishes with Two-Stage Encryption. Our software uses both AES symmetric encryption as well as RSA asymmetric encryption to avoid file size limitations per AES(Brent).

Per Wireshark analysis, files are encrypted before being sent to the other party using TCP. The potential market for the product is very promising, because we solved the problem of how to privately transfer files.

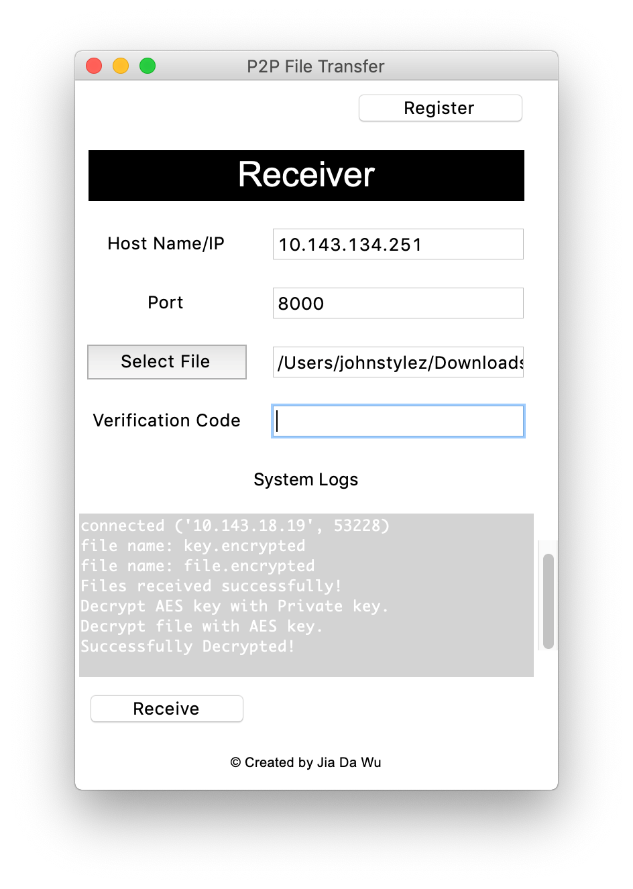
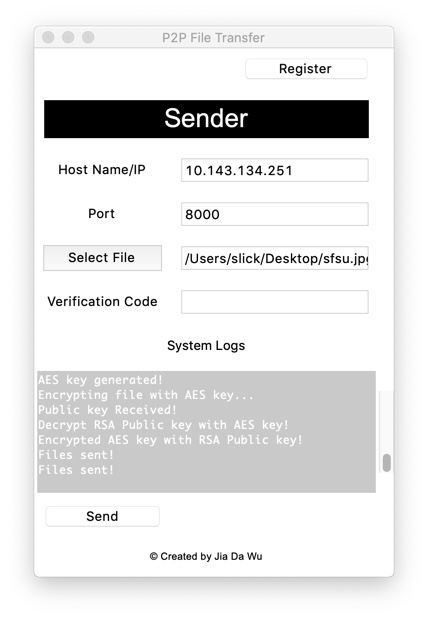
**Core Concept**

**Problem**—How can we share files (intellectual property, contracts, employee and/or customer information, medical records, other documents you need to keep confidential) securely, if the Internet is an open space and networks are subject to snooping?

**Problem to worsen**—“The encryption software market size is expected to grow from USD 3.87 Billion in 2017 to USD 12.96 Billion by 2022, at a Compound Annual Growth Rate (CAGR) of 27.4%. The demand for encryption software is likely to be driven by various factors, such as proliferation in the number of cyber-attacks and the stringent government regulations and compliances that mandate the adoption of encryption among various verticals”(Encryption Software Market by Component, MarketsandMarkets.com).

**Solution**—Encrypt files so only the intended party has access to see the shared files. Authenticate network users to keep unauthenticated users out.

**Our Approach**—We created software that incorporates our solution to solve the problem articulated above. Our software enables person-to-person secure file sharing and is called, P2P Transfer. It is built with email delivered download options of “Receiver” or “Sender” files, according to using your computer to either send or receive the file(s) to be shared. P2P File Transfer software uses three core concepts to securely share files: The first is, a Registration PIN; it allows you to establish a connection with our P2P File Transfer software. The second is, Two-Factor Authentication; it verifies you are who you claim to be via the Google Authenticator App, which generates a Verification Code to your cell phone after you scan a QR Code to complete the registration process. The third is, Two-Stage Encryption; our product uses both AES symmetric encryption and RSA asymmetric encryption to avoid file size limitations per AES(Brent) and was created in Python. Below are our P2P File Transfer software System Logs to verify encryption. For complete interface of P2P File Transfer software see Advanced User Guide section. Since the files are encrypted between the “Sender” and “Receiver”, the files sent between them are completely private as they move across any network including the open internet. Our private connection is verified by diagnostic testing of P2P File Transfer software via Wireshark diagnostics.



**Our Software**—Our P2P File Transfer software is a prototype with the requirement to meet in person to provide/receive the Host Name/IP, Port Name, and Registration PIN— alternatively this information can be mailed. Additionally, a cell phone with the Google Authenticator App and a working email address will be needed to use our software.

**Additional P2P File Transfer Software Benefits**—P2P File Transfer software can deliver secure file sharing capacity without internet connection required as it functions on a LAN. P2P File Transfer software does encryption internally so you do not have to worry about storing your block encryption key(s). Since P2P File Transfer software encrypts your files, there is no need to worry about snooping on the network any longer.  Two-factor authentication can thwart middle-man-attacks as only you will have your cell phone and know its passcode to use it. User friendly interface makes secured file sharing a simple process.

**Future Enhancements**—Due out next semester, we would make our software accessible via a web page rather than email download.  Then we would use Apache Cordova or Ionic Framework(Verma) to mobilize our web page and present our web page as an App to users. It would remove the need for authenticating the user, emailing the software for download, and enable billing of P2P File Transfer on a pay as you go basis available on your device. Additionally, we would move towards strictly using asymmetric encryption to eliminate the need to meet in person as is currently required with our current software prototype. We would also explore delivery of the public key by sharing it after the # in our link to our web page, for example: P2PTransfer.com#public-key-goes-here-for-sender-to-use(Sneddon, Joey) to avoid having to find a place host your public key.

**Our Team**—Jia Da Wu –Software Development and How to get started Advanced User Guide Lead, Keshawn Gosiengfiao— Wireshark Diagnostic Testing Lead, Lilia Barajas Lopez—Core Concept and How to get started Beginning User Guide Lead, Victor Llano Mariota— Executive Summary Lead (where not listed as Lead, other team members supported Lead).

**How to get started: Beginner User & Advanced User guides**

**Beginner User Guide:**

People acting as “Receiver” and “Sender” will need to meet in person to provide/receive the Host Name/IP, Port Name, and Registration PIN. If people acting as “Receiver” and “Sender” cannot meet in person, we recommend the “Sender” request the details in writing so the “Receiver” can physically mail you the needed information via FedEx. Cell phone needed for authentication and must have QR Scanner app present. Email that is working is required to receive P2P File Transfer software download files.

**How to Get Started as “Receiver”:**

1.      Download program file: Open file, select Receiver folder, and download it(“t[kinter](https://docs.python.org/3/library/tkinter.html#module-tkinter)

        — Python interface to Tcl/Tk”).

2.      Push “Register” button, enter Registration PIN provided(“PyOTP - The Python

        One-Time Password Library”; “PyQRCode 1.2.1”).

3.      Scan QR Code with phone to receive required “Verification Code”(“PyOTP - The

        Python One-Time Password Library”; “PyQRCode 1.2.1”).

4.      Browse for Save File location by pushing “Save File” button.

5.      Enter “Verification Code” generated from google authenticator to your cell

        phone(“PyOTP - The Python One-Time Password Library”).

6.      Push “Start Server” button.

**How to Get Started as “Sender”:**

1.       Download program file: Open file, select Sender folder, and download it(“t[kinter](https://docs.python.org/3/library/tkinter.html#module-tkinter)

         — Python interface to Tcl/Tk”).

2.       Push “Register” button, enter Registration PIN provided(“PyOTP - The Python

         One-Time Password Library”; “PyQRCode 1.2.1”).

3.       Scan QR Code with your phone to receive required “Verification Code”(“PyOTP -

         The Python One-Time Password Library”; “PyQRCode 1.2.1”).

4.       Enter the Host Name/IP and Port Name provided(“t[kinter](https://docs.python.org/3/library/tkinter.html#module-tkinter) — Python interface to

         Tcl/Tk”).

5.       Push “Select File” button and select the file you want to send(“t[kinter](https://docs.python.org/3/library/tkinter.html#module-tkinter) — Python

         Interface to Tcl/Tk”).

6.       Enter “Verification Code” generated from google authenticator to your cell

phone(“PyOTP - The Python One-Time Password Library”; “PyQRCode 1.2.1”).

7.       Push “Send” button; file(s) will be encrypted internally and sent to the

“Receiver”(“Cryptography 2.6.1”).

**Advanced User Guide:**

**Step 1**

Administrator/Receiver:

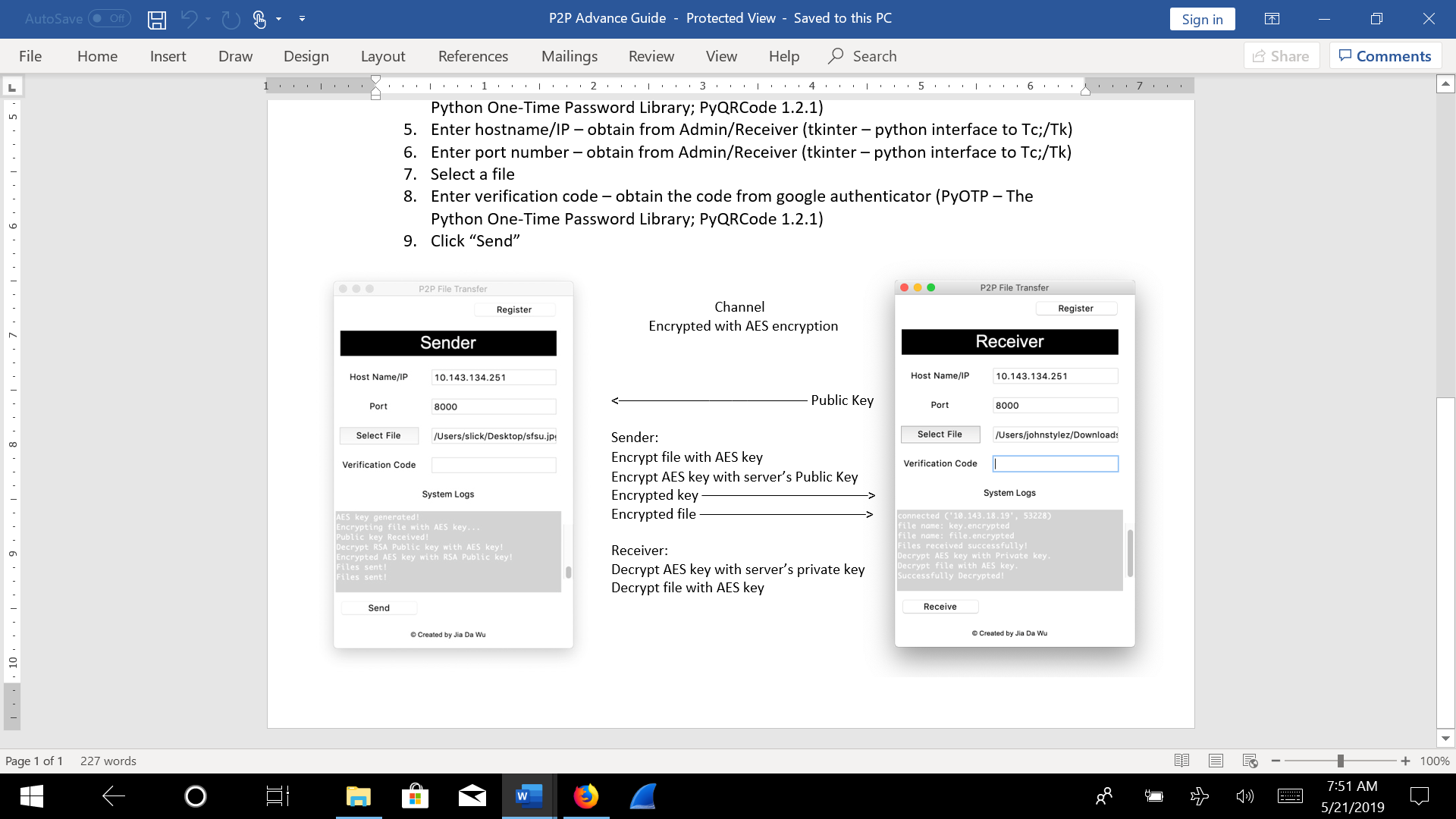
Software, 6 digits Passcode – for authentication register, QR code Register, Hostname/IP, Port Number.

**Step 2**

1. Download the software.
2. Open software(“tkinter – python interface to Tc;/Tk”).
3. Register – obtain the code from Admin/Receiver(“PyOTP – The Python One-Time Password Library”; PyQRCode 1.2.1”).
4. Scan QR code with google authenticator to complete the registration(PyOTP – The Python One-Time Password Library; PyQRCode 1.2.1).
5. Pick a location to save the file.
6. Enter verification code – obtain the code form Google Authenticator(“PyOTP – The Python One-Time Password Library”).
7. Click “receive”.

**Step 3**

1. Download the software.
2. Open software(“tkinter – python interface to Tc;/Tk”).
3. Register – obtain the code from Admin(“PyOTP – The Python One-Time Password Library”; “PyQRCode 1.2.1”).
4. Scan QR code with google authenticator to complete the registration(“PyOTP – The Python One-Time Password Library”; “PyQRCode 1.2.1”).
5. Enter hostname/IP – obtain from Admin/Receiver(“tkinter – python interface to Tc;/Tk”).
6. Enter port number – obtain from Admin/Receiver(“tkinter – python interface to Tc;/Tk”).
7. Select a file.
8. Enter verification code – obtain the code from google authenticator(“PyOTP – The Python One-Time Password Library”; “PyQRCode 1.2.1’).
9. Click “Send”



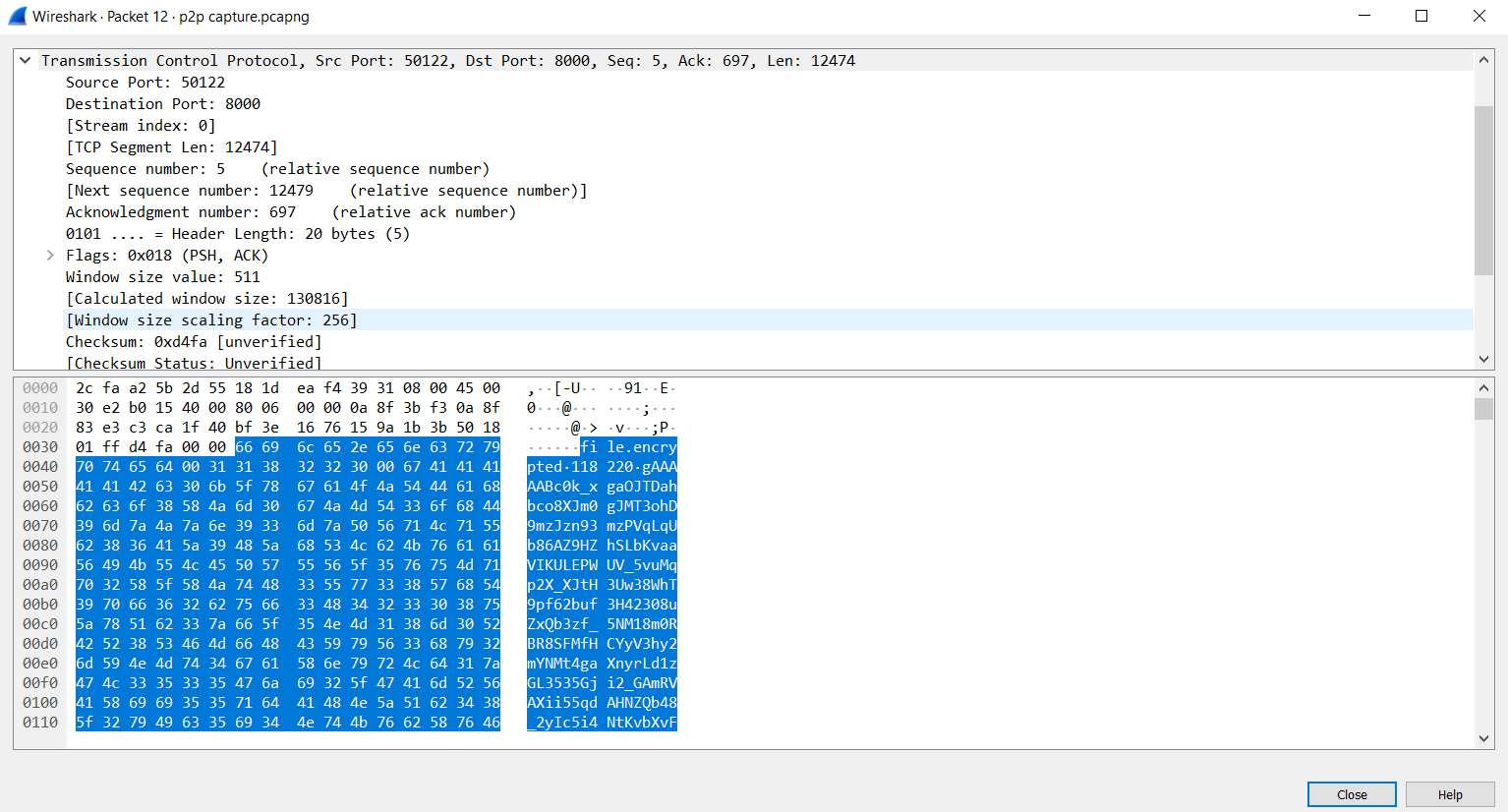
**Wireshark Evidence**

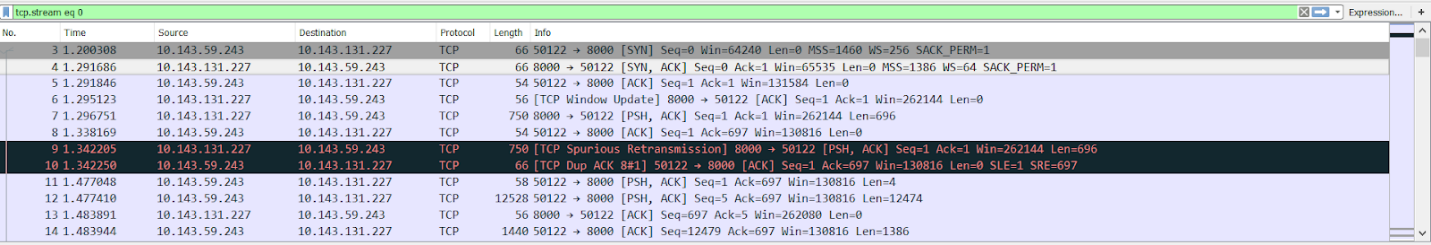
**Application Layer:**

* The software itself is the application layer.
* Since the transmission is encrypted, Wireshark evidence for the Application Layer is not available.

**TCP Layer:**

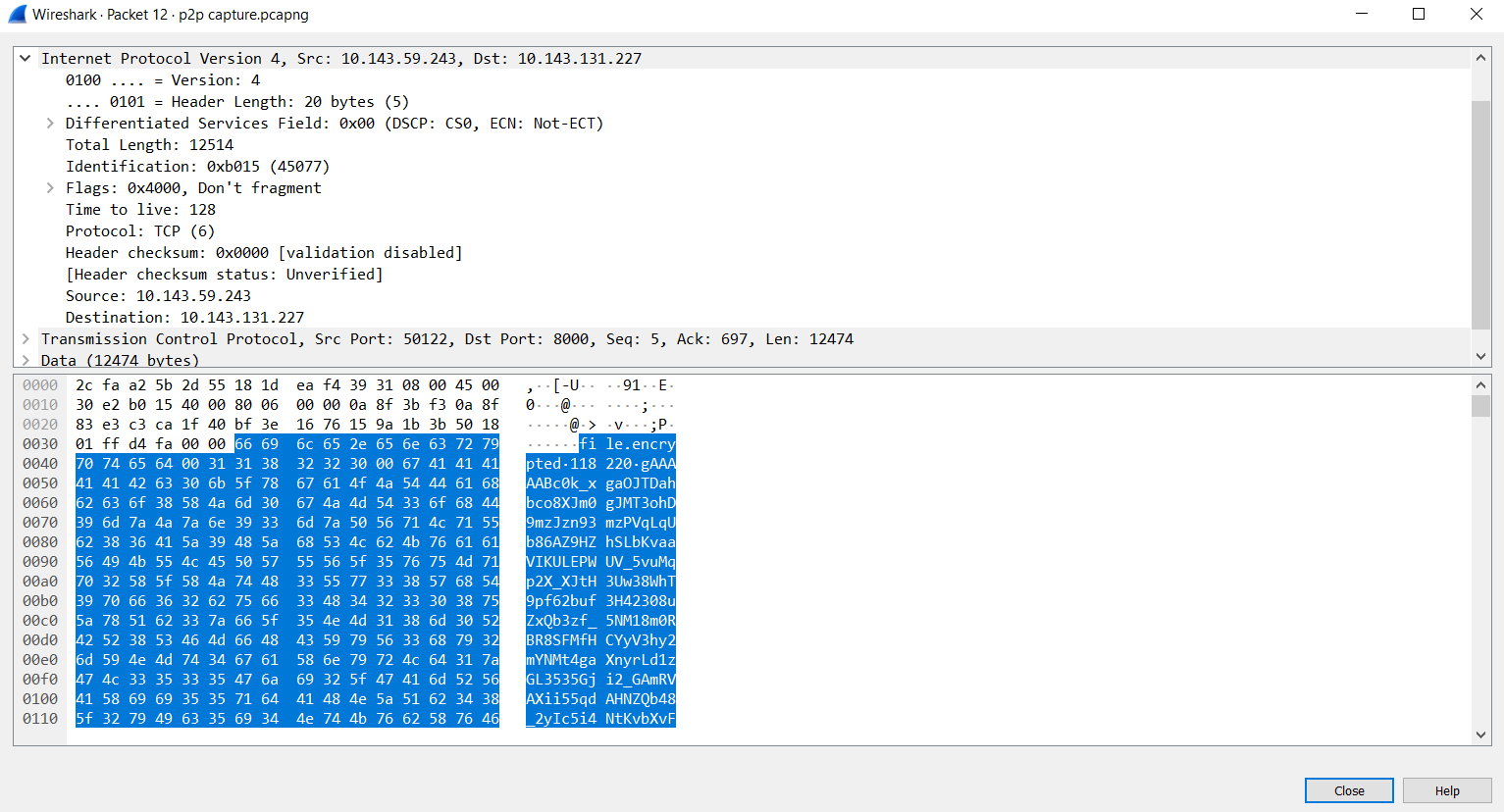
* TCP layer is shown by using TCP to transmit data from one place to another.
* TCP is used from beginning to end for the File Sharing process.
* Proof in Lines 3-113 in Wireshark.

****

****

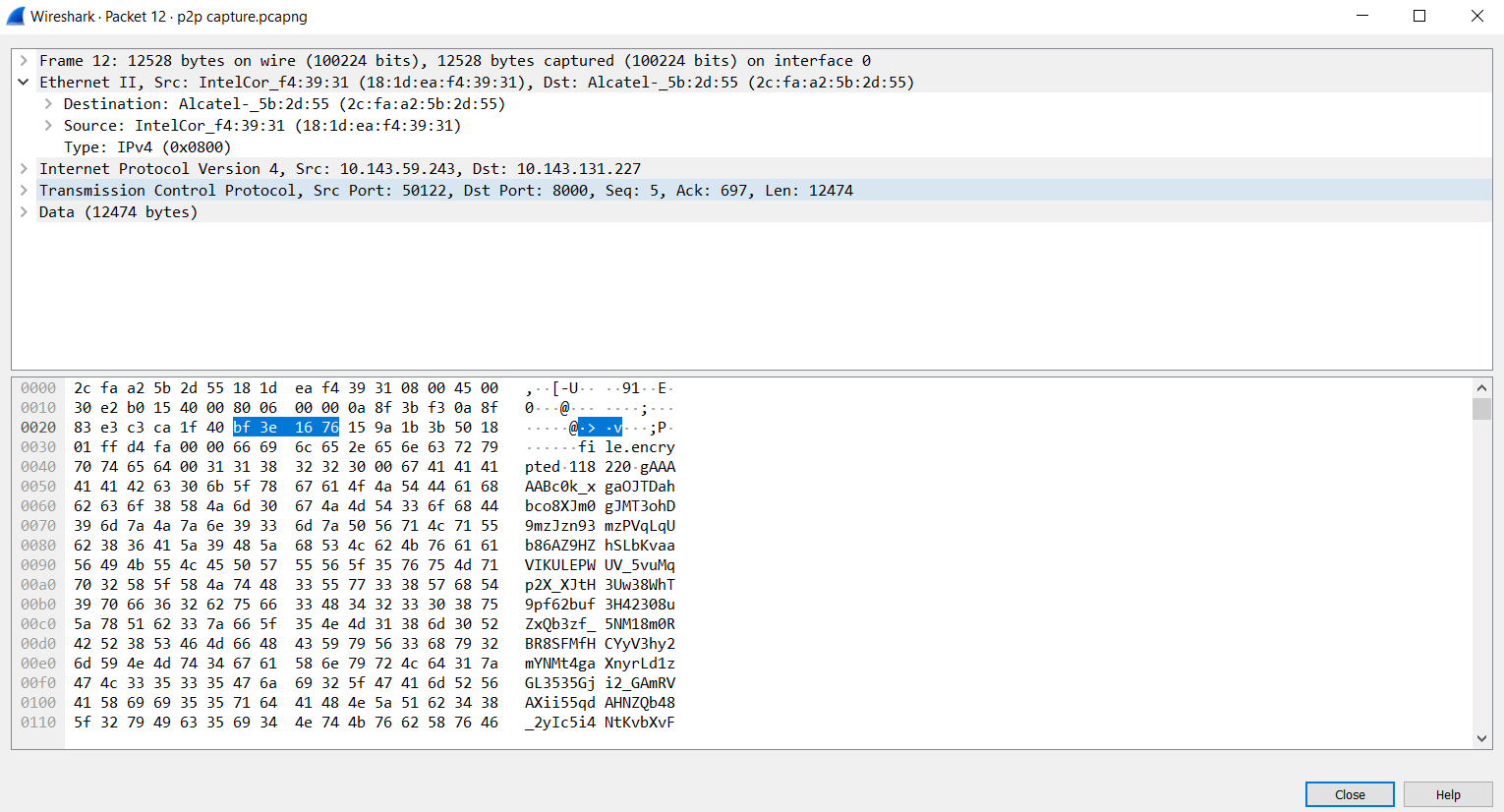
**IP Layer:**

* Sender IP: 10.143.59.243.
* Destination IP: 10.143.131.227.
* (Evidence in capture. Sender and Destination address are found in multiple lines).

****

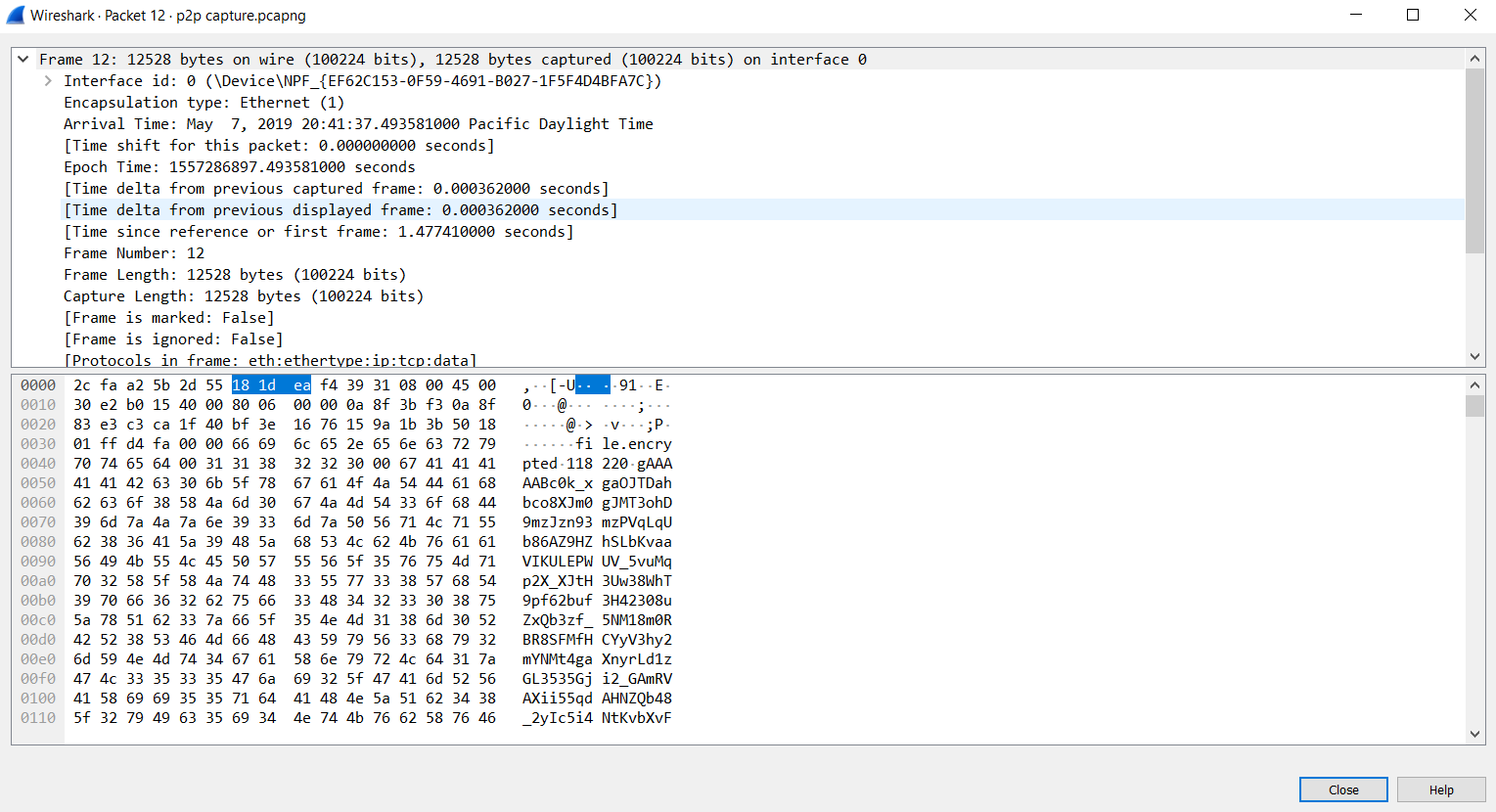
**Data Link:**

* Wi-Fi.
* Ethernet.
* Ethernet 2.

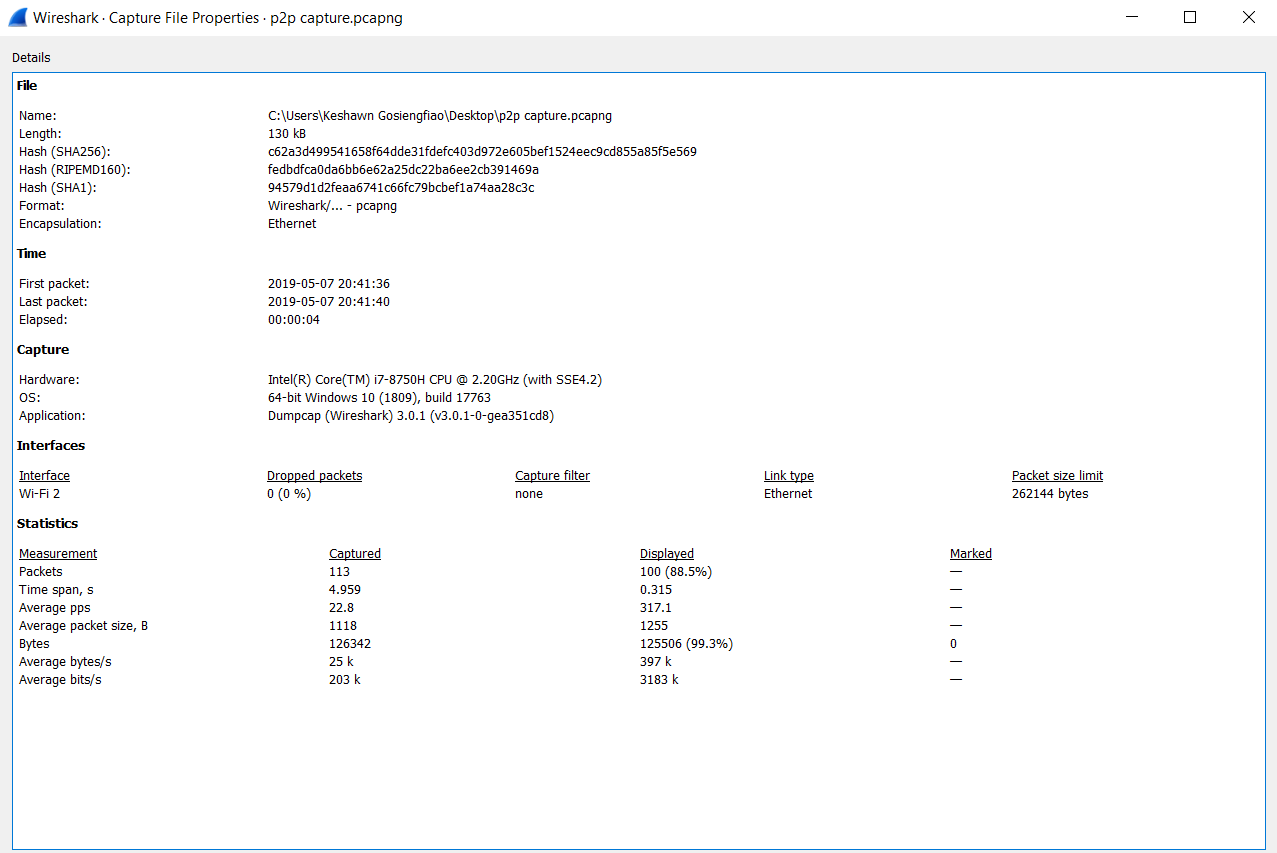
****

**Physical Layer:**

* Mac or Windows Computer with network capabilities (Wi-Fi or Ethernet).
* Mac Address/Physical Evidence is shown in capture.

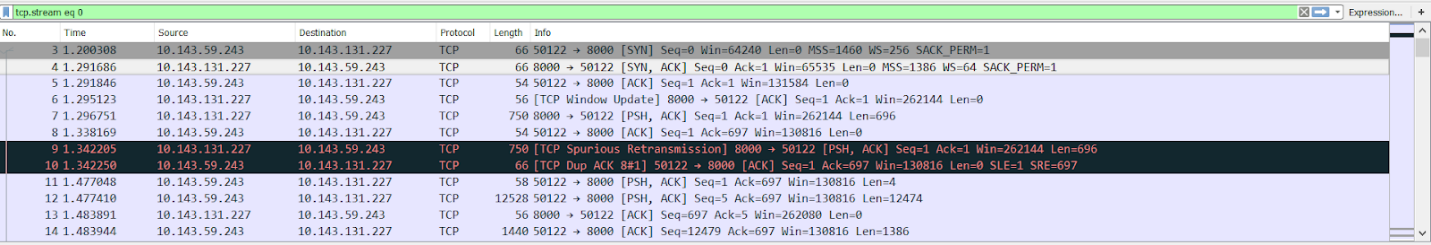
****

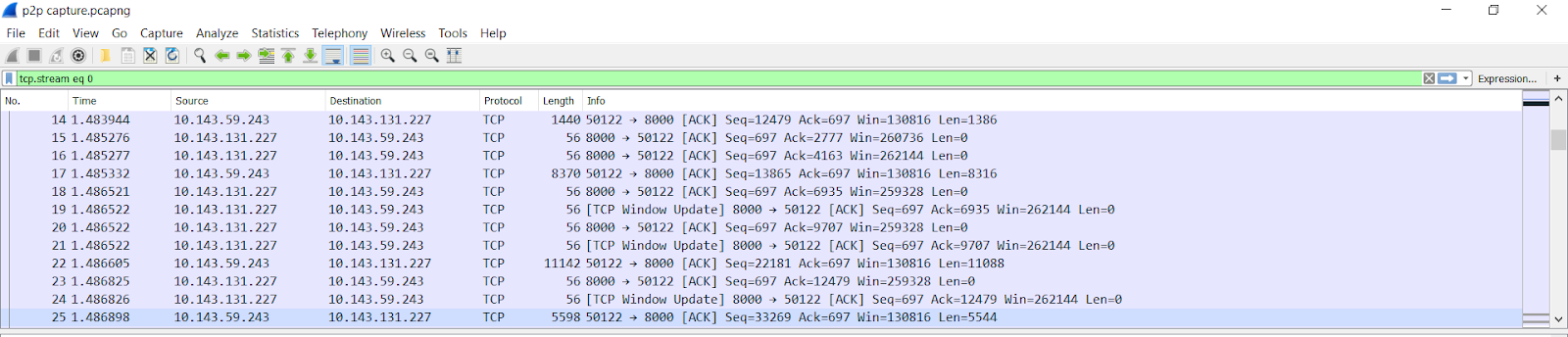
**Overview:**

****

**Expected Protocol:**

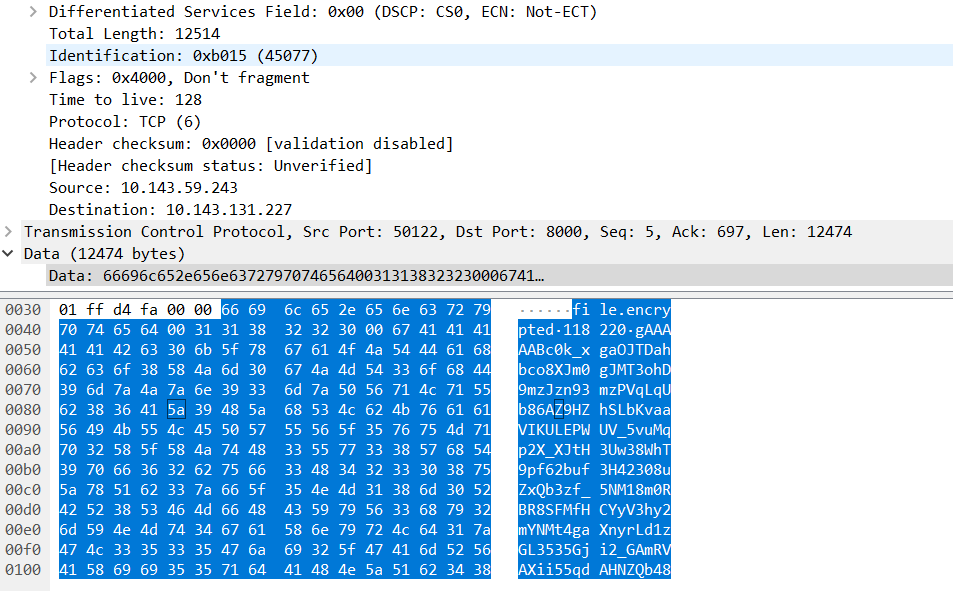
* Expected Protocol is TCP for File Transmission (Lines 3-113).

****

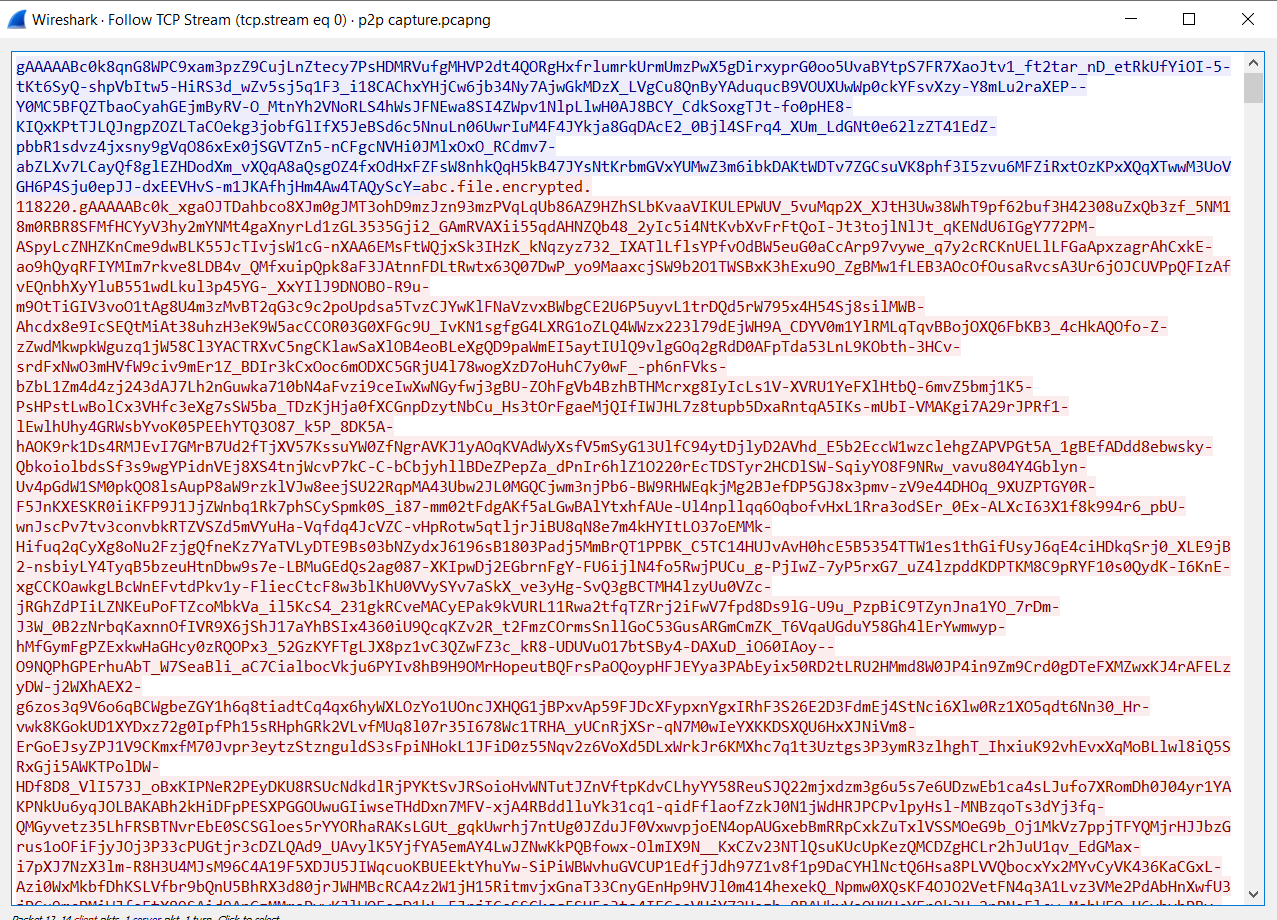
****

**Encryption:**

* AES symmetric encryption and RSA asymmetric encryption:

****

**Encryption Continued:**

****

# References

“A Deep Dive on End-to-End Encryption: How Do Public Key Encryption Systems Work?.”  *Surveillance Self-Defense*. 29 November 2018. <https://ssd.eff.org/en/module/deep-dive-end-end-encryption-how-do-public-key-encryption-systems-work>. Accessed on April 12, 2019.

Brent. “Encryption and Decryption in Python.” *Nitratine*. 18 December 2018. <https://nitratine.net/blog/post/asymmetric-encryption-and-decryption-in-python/>. Accessed on April 12, 2019, used Brent comments to resolve file limit size of AES.

# “Cryptography 2.6.1.” 27 February 2019. *Python Software Foundation (US).* [*https://pypi.org/project/cryptography/*](https://pypi.org/project/cryptography/). Accessed on April 13, 2019.

“Encryption Software Market by Component (Solution and Services), Application (Disk Encryption, File/Folder Encryption, Communication Encryption, and Cloud Encryption), Deployment Type, Organization Size, Vertical, and Region - Global Forecast to 2022.” *MarketsandMarkets.* October 2017. <https://www.marketsandmarkets.com/Market-Reports/encryption-software-market-227254588.html>. Accessed on April 9, 2019.

# “How to: Use Signal for Android.” *Surveillance Self-Defense*. 9 May 2018. <https://ssd.eff.org/en/module/how-use-signal-android>. Accessed on April 12, 2019.

“How to: Use Signal on iOS.” *Surveillance Self-Defense*. 9 May 2018. <https://ssd.eff.org/en/module/how-use-signal-ios>. Accessed on April 12, 2019.

“PyOTP - The Python One-Time Password Library.” 2015. *PyOTP*. PyOTP contributors Revision 425a0ec4. <https://pyotp.readthedocs.io/en/latest/>. Accessed on April 15, 2019.

“PyQRCode 1.2.1.” 19 June 2016. *Python Software Foundation (US).*

<https://pypi.org/project/PyQRCode/>. Accessed on April 17, 2019.

Sneddon, Joey. “Firefox Send is a Free, Encrypted File Sharing Service.” 15 March 2019. *omg! Ubuntu!*. <https://www.omgubuntu.co.uk/2019/03/firefox-send-encrypted-file-sharing>. Accessed on April 10, 2019.

“t[kinter](https://docs.python.org/3/library/tkinter.html#module-tkinter) — Python interface to Tcl/Tk.” 10 May 2019. *Python Software Foundation (US).* <https://docs.python.org/3/library/tkinter.html>. Accessed on April 12, 2019.

Verma, Samer. ISYS 565-Managing Enterprise Networks: “Cloud, Mobile, IT.” 7 May 2019. San Francisco State University, College of Business. Lecture.