CS6002 Distributed Systems

Mobile Devices and Security

Coursework 2

Report

By: Lauren Spruce

ID: 18011848

Table of Contents

Table of figures 3
Tables 3
Introduction 4
Importance of Mobile Security 4
Types of Mobile Security Threats 4
Mobile Device Security Solutions 5
Importance of Encryption 5
Advanced Encryption Standard 5
How AES works 6
AES Structure 6
Advantages of AES 7
Disadvantages of AES 7
Importance of VPNs 7
How VPNs work 7
Advantages of VPNs within Mobile Devices 8
Disadvantages of VPNs within Mobile Devices 8
Example of Mobile VPN- Nord VPN 8
Importance of Mobile Proxy Services 9
Advantages of Proxy Services 9
Disadvantages of Proxy Services 9
Example of Mobile Proxy Service- SmartProxy 10
Implementation 11
Mobile Application: AES Encryption 11
General Software Description 11
System Requirements 11
Software Requirements 12
User Interface 12
Results and Testing 13
Encrypting Text 13
Decrypting AES 14
Entering incorrect password 15
Results and Observations 16
Encryption Method 16
Decryption Method 17
Decode Button 17

Encode Button 18
Generating Keys 18
Conclusion 19
References 20

Table of figures

Figure 1: User interface of application 12
Figure 2: Encrypting text within application 13
Figure 3: Decrypting text within application 14
Figure 4: Entering incorrect password 15
Figure 5: Code: Encryption method 16
Figure 6 Code: Decryption method 17
Figure 7: Code: Decode button method 17
Figure 8: Code: Encode button method 18
Figure 9: Code: Generating code method 18

Tables

Table 1: 4x4 layout for AES 6
Table 2: Other ciphers, standard line format of bytes 6

Introduction

For my research on current cybersecurity issues, I will be reviewing the security on mobile devices. I will be discussing the importance of mobile security, followed by some examples of security threats, and detailing the solutions currently implemented to minimise the negative impacts different threats may have to sensitive personal data. This will be followed by my own implementation, which relates to AES encryption within Mobile Security.

Importance of Mobile Security

Mobile phones have become a necessity in modern day life. In 2020, there was a projected "*14.02 billion mobile phones worldwide*". (O'Dea, 2020) From keeping families and businesses connected and continuing to save many lives in emergencies, this invaluable portable device as endless functionality which has been successfully implemented into all our lives.

With the many functionalities continuing to expand, like mobile banking, continuous internet access and GPS technology, this also means that the demand and importance for mobile security also drastically increases, to protect user's and businesses data. As there are many different functionalities regarding this advanced technology, it shows that mobile security needs to cover and secure a wide range of threats, to counteract and reduce data theft.

Types of Mobile Security Threats

Due to the increased use of mobile phones within the public, this has increased the malicious ways in which many user's personal data may be threatened or stolen.

Some of these types of threats include:

- **Physical loss of mobile phone**

A user may misplace their phone or have it stolen from their person while travelling, this type of threat allows the thief or other individual complete hardware access. This can be threatening to user's and businesses with mobile phones containing invaluable personal and confidential data.

To minimise the impact of a lost phone, users are advised to use a suitable password to protect their data. Also, apple allows for phone location as an attempt of recovering the phone.

Apple has also *"preinstalled tools that allow you to remotely wipe the data on your iPhone and ensure no one can steal your information if it is lost or stolen."* (Dargin, 2017) This feature proves useful for business phones which might contain highly confidential data.

- **Network threats**

Network threats are where *"cybercriminals can steal unencrypted data while people use public Wi-Fi networks."* (Gontovnikas, 2017) This proves to be very dangerous as many user's have their mobile phones automatically connect to free WIFI through mobile settings, other's may not realise the dangers of connecting. This happens when the hacker spoofs the web server's security certificate, therefore many will not realise that they are connecting to a fake Wi-Fi set up by a third party. A study carried out in 2016 *"found that 70% of people connected to unsecured, public networks."*

(Gontovnikas, 2017)This shows the risk that many user's data could easily have been recorded and used by another third party, showing the importance for more education of the dangers of public networks.

A way of reducing the threat is to provide more education, to inform users not to connect to public networks if they seem counterfeit, also the importance of using different passwords for different accounts. This is to ensure that if one password is stolen other personal accounts will be secure. Another way of preventing data being stolen is to use a VPN while accessing the internet using 3G,4G and 5G.

- **Mobile applications**

Another threat is apps created with the sole purpose of stealing data from a user's device. These apps seem harmless to a user, as they are usually apps with an intended purpose to attract users. Once downloaded, the app will ask for permission to access files for the user to interact with the app. Most user's normally overlook these permissions and due to this, leaves their personal data at risk of being analysed and sent to a third party.

- **Phishing**

One threat which is continuously increasing is phishing*. "Phishing is a method of trying to gather personal information using deceptive e-mails and websites."* (Fruhlinger, 2020) As many user's keep up to date with personal and corporation emails on the daily, this is an easy target for cybercriminals. They will pretend to be the user's bank or for example, amazon, send the user a message notifying them that they need to update their personal information, including a fake link. This link will redirect the user to a fake website in which they will request that the user fills out all their personal details and sometimes even bank details. This data is then stored and used by a third party.

Mobile Device Security Solutions

Importance of Encryption

Encryption is one of the most effective ways of keeping your data secure on your mobile device. *"Encrypting the data takes security to a much further step. It makes the information on your phone unreadable."* (Dargin, 2017) This is important in case the mobile device is lost or stolen, if the data is encrypted, third parties will not be able to access the valuable data stored within it. This also applies to using any network, if data being sent and received is encrypted, it prevents third parties from stealing private data.

Advanced Encryption Standard

The most effective and widely used mobile device used is currently Advanced Encryption Standard (AES), this has now become the global standard of encryption. *"Advanced Encryption Standard was introduced by NIST (National Institute of Standard and Technology) in the year 2001."* (T, 2020) Since then, it considered the most efficient and secure way of protecting mobile device data.

AES is a symmetric block cipher, this cipher is used to encrypt electronic data and decrypt it, to keep user's data safe. It can be implemented functionally into software and hardware to keep data secure.

How AES works

AES works by all the data being split into chunks. The standard block size of AES is 128-bits. Once data has been split, the next stage is key expansion. This involves *"taking the initial key and using it to come up with a series of other keys for each round of the encryption process."* (Lake, 2020) The next stage is to add the initial key to the original data, which is done using a XOR cipher. This XOR cipher is an *"additive encryption algorithm."* (Lake, 2020) Binary is used to add both the initial key and data together. The substitution of bytes is then used in guidance of an existing table, this just means to replace certain bytes with it is matching one from within the table. The algorithm then allows for each row to be shifted to the left or the right. After shifting, the columns are mixed using a mathematical equation to further increase security.

Now, the round key and initial key are now reintroduced into the mixed columns. This process is continued with many more rounds, continuing to repeat the process depending on the key size of AES. *"AES has key sizes of either 128, 192 or 256-bits,"* (Lake, 2020) showing that the algorithm has many different stages to fully encrypt data. This gives the user the option of how much security they want to implement, the higher number of bits, the increased level of security.

AES Structure

AES gets arranged into a grid, unlike most ciphers which prefer to do so in a line. This will be a 4x4 grid for 128-bits.

Table 1: 4x4 layout for AES

Byte 00	**Byte 04**	**Byte 08**	**Byte 12**
Byte 01	**Byte 05**	**Byte 09**	**Byte 13**
Byte 02	**Byte 06**	**Byte 10**	**Byte 14**
Byte 03	**Byte 07**	**Byte 11**	**Byte 15**

Table 2: Other ciphers, standard line format of bytes

Byte 00	**Byte 01**	**Byte 02**	**Byte 03**	**Byte 04**	**Byte 05**	**Byte 06**	**Byte 07**	**Byte 08**	**Byte 09**	**Byte 10**	**Byte 11**	**Byte 12**	**Byte 13**	**Byte 14**	**Byte 15**	**Byte 16**

Advantages of AES

- AES can be used on both software and hardware, showing it is very versatile.
- It is implemented in a wide range of applications, for example, messaging applications like *"WhatsApp and Signal."* (Lake, 2020)
- It uses 3 key lengths which include *"128-bits,192-bits and 256-bits."* (T, 2020)

Disadvantages of AES

- *"Hard to implement with software,"* (RF Wireless World, n.d.) as it can affect a devices performance as it is a complex algorithm.
- Every block of data is always encrypted in the same format, so therefore the algorithm could be seen as easily breakable.

Importance of VPNs

Due to the many network threats your mobile device is constantly facing, nowadays VPNs are seen as an essential. Many people have heard of VPNs for laptops and desktops; however, the popularity of mobile VPNs is increasing daily.

VPN stands for "virtual private network." It provides two main advantages, which are:

- Security

This protects all a user's personal information while it is being received and sent over the internet.

- Privacy

It provides a user with more privacy as it masks *"things like your IP address, location, and search history, to keep them from being tracked by websites, internet browsers, cable companies, internet service providers (ISPs), and others."* (Johansen, 2020) Like with AES encryption, some VPN's will be able to provide 256-bit encryption for data, to make them highly secure and protected.

How VPNs work

VPNs rely on encryption and remote servers to mask a user's data. Encryption will encrypt all the user's data into an unreadable state for any third party, this data can be decrypted by the user with a key. *"Tunnelling protocols are then used to create encryption "tunnels" within your Wi-Fi or cellular network."* (Kaspersky, 2021) This hides the user's data from other user's who may be using the same network, which is important within mobile devices as a user may connect to many different networks whilst travelling. *"Remote servers act as the destination for your encrypted data tunnel. Your data gets shipped out to VPN servers in other countries to help anonymize your identity."* (Kaspersky, 2021) This will hide your location and therefore act as a middleman within the user's connection between the user and the network, this is crucial for mobile devices as it prevents third parties from discovering your IP address as this can be easily accessible from simply connecting to an unsecure network.

Advantages of VPNs within Mobile Devices

While using a public network, third parties will not be able to view encrypted data. This is due to *"an 'encryption key' to make sense of it, and brute force attacks."* (Kaspersky, 2021) This would take a long time to encrypt, so therefore this is a huge benefit for mobile devices as user's data is secure on public networks, making it more secure and safe to use or connect to them whilst travelling.

Another benefit is for businesses, due to the COVID pandemic, a lot of businesses need to work remotely. Therefore, *"Corporate data breaches are less of a risk during remote working."* (Kaspersky, 2021) Many businesses require its employees to use VPNs to access company data, as it increases the company's overall security and will therefore reduce data breaches. This is due to VPNs connecting to private secure servers. Many employee's will use mobile devices to answer emails while travelling, so therefore its vital companies implement VPNs into their company's overall security.

Disadvantages of VPNs within Mobile Devices

- VPNs may decrease your internet speed, as *"they reroute and encrypt your internet connection."* (Johansen, 2020) This is a particular disadvantage for mobile devices as they use wireless connection which is typically slower than wired connection through ethernet.
- They are quite hard to setup, this could be potentially worse for mobile devices, as most mobile users do not have the knowledge of VPNs, essentially beginners. This could be due to the lack of awareness, as you need to have a deep understanding of encryption to choose the safest and highest encryption quality. Therefore, many users could opt out of downloading one.

Example of Mobile VPN- Nord VPN

A popular example of a mobile VPN is Nord VPN. NordVPN claims to have *"the fastest VPN service on the planet."* (NordVPN, 2021) Therefore, it is a highly sought-after choice for individuals and employees to have the fastest internet speed possible, whilst having the benefit of added security. *"Thanks to our revolutionary NordLynx protocol, NordVPN is the fastest virtual private network provider out there."* (NordVPN, 2021) This shows that NordVPN have created their own protocol, which shows that the company excel in protecting and keeping user data safe.

NordVPN works across multiple platforms, Windows, Mac for computers, and android and iOS for mobile devices, showing that users can keep a wide range of devices secure. *"A single account lets you connect up to 6 devices at the same time."* (NordVPN, 2021) This makes it easy and simple for inexperienced VPN users to keep all their devices safe. NordVPN is simple and easy to use, with a simple user interface for mobiles making it ideal choice for all individuals, as it does not require encryption knowledge or complex steps to set up.

Importance of Mobile Proxy Services

A proxy service can be used on a tablet or mobile phone. They are important as they *"connect to such proxies to mask your real IP address with the device's one."* (Infactica, 2020) This means that the user's IP address is hidden from third parties so therefore they have an increased level of security while accessing public networks. *"The network of mobile proxies consists of real 3G or 4G connections that are assigned to certain portable gadgets by a mobile operator."* (Infactica, 2020) This means that the network is connected by devices that individuals use daily, this means that all the IP addresses are real, the devices are connected from around the world so you can choose to mask to your mobile IP to another country and be undetected.

Proxies are like VPNs, essentially, they mask your IP address on one mobile device and use another IP address from another device. They *"intercept incoming and outgoing requests to see if it can fulfil the requests itself. If not, it then forwards the request to the real server."* (Webopedia Staff, 2021) This is usually done by done through a firewall. Proxies reduce the network traffic making it quicker to access data, improve overall performance and they also increase security.

Advantages of Proxy Services

- Blocking sites

Proxies can be used to block certain sites from being accessed from the mobile device. This can be an effective way for companies to remove unnecessary sites or ones that go against the companies' policies.

- Privacy

Proxies allow for secure and private browsing, as they *"avoid getting inundated with unwanted ads or the collection of IP-specific data."* (fortinet, 2021) This means that you are completely hidden using that mobile device and you have complete privacy without third parties collecting your data.

- Security

Proxies act as firewalls between the mobile device and the internet, this is an advantage as third parties cannot access your IP address which they can use to steal private data. It sends a request to the proxy, which then sends a request to the website you are trying to access. This will result in masking your identity, as the proxy is acting like a 'middleman.'

Disadvantages of Proxy Services

- Proxy provider watching your activity

"Some providers might still be able to see and monitor your activity online." (Buskirk, 2019) This means that while accessing a network and browsing on a mobile device, you are not completely hidden as the provider will be able to view any activity completed while connected.

- Cost to setup

For small businesses, the cost of setting up a proxy is high there are lots of steps involved to setup. "*Besides installation there are various other expenses involved."* (Roomi, 2020) The maintenance of a proxy service can be costly as there are risks of it going down, also costs for expansion needed to be taken into consideration.

- Incompatibility

Sometimes, the proxy service may not be compatible with the network which you are currently using. This means that configuration may be needed, or *"go with a proxy that completely matches with the network you are using."* (Roomi, 2020) This is a disadvantage as it would take a user a lot of time and research to find a perfect match which is compatible for their own network, which may put off a lot of user's trying to install a proxy.

Example of Mobile Proxy Service- SmartProxy

An example of a mobile proxy I have researched is SmartProxy. I found that this was the best option for beginner users, as the prices are affordable and overall, it is simple to configure and setup on a mobile device. *"Smartproxy offers multiple plans for both residential and data centre proxies that vary based on the traffic allowance."* (Sharma, 2020) This allows for users to choose how much data they need based on their lifestyle and workload.

One of SmartProxy's main features is it is *"rotating residential proxy,"* (Sharma, 2020) which *"offers over 40 million IPs from over 195 locations around the world."* (Sharma, 2020) This allows for a large variety of IP's which are accessible to the user, allowing the user to mask their real IP with relative ease, this therefore makes it harder to track the user.

Implementation

Mobile Application: AES Encryption

General Software Description

I will be implementing the AES algorithm for encryption and decrypting password's using the programming language Java. While researching AES Algorithm, I thought it would be interesting to try and implement this encryption method using a programming language I was familiar with, Java.

To relate AES encryption to my chosen topic, Mobile Devices and Security, I chose to program a mobile application. This allows users to encrypt and decrypt text using AES, within android studio. I felt this was important as it shows the encryption being performed from the mobile emulator, I used a google pixel 4 to demonstrate this. A key size of 256-bits was implemented to generate a securer encrypted text.

The application allows for users to input text within the field box provided. It shows a hint to help the user navigate around the page with ease. Also, the user must enter a password for the test they would like to encrypt, this will further increase the security, therefore the text cannot be decrypted by anybody who does not know the password.

Once the user has inputted the initial text and password, they will need to interact with the 'Encrypt' button to encrypt the initial text into AES. To decrypt the encrypted text, the user will need to interact with the 'decrypt' button to decrypt back to the initial text. Note, that the same password will be needed to decrypt the text, if changed, the user will be presented with a 'Incorrect Message' message.

Here is a summary of what my application delivers:

- Mobile Application using android studio and emulator
- Fully functioning encrypt and decrypt buttons
- Fully functioning interactive text fields
- Validation on password, to prevent unauthorised users from decrypting text

System Requirements

- **Disk Space:** Minimum 1GB HDD or SSD
- **Processor:**

  Intel: i3 Processor, or higher

  AMD: Ryzen 3 1300X or higher

- **Operating System:** Windows 7 or preferably the latest version, Windows 10
- **Internet Connection:** Not required

Software Requirements

Software requirements to run the application successfully, user's will need this required software installed on their PC:

- Android studio

User Interface

I will now present my main application user interface of my AES application. I gave a brief introduction to the application, to give user is a clearer description of how to interact with the application. Also, to help aid the user's I added hints on the text field's so the user knows which text field to write the correct information. I also added an encrypt and decrypt button for the users to interact with.

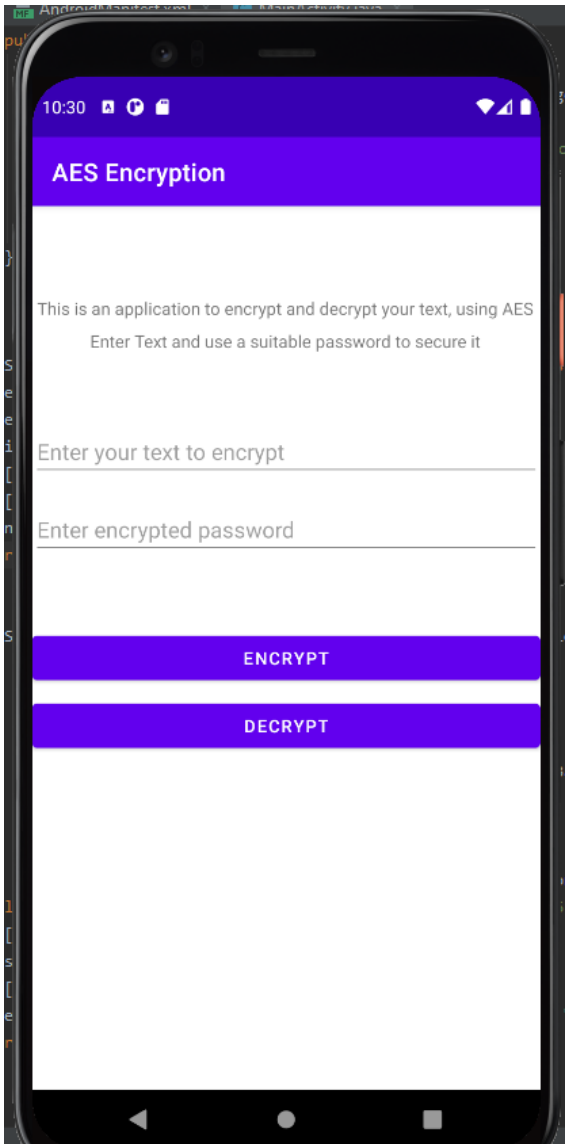

Figure 1: User interface of application

Results and Testing

Encrypting Text

First, I will be showing the encryption button and text field working. I will enter the initial text within the encrypt text field and then adding a password for the text within the password text field. Then, I will interact with the encrypt button to receive my initial text encrypted to AES.

Once all suitable information has been entered, the following text will be encrypted, as follows:

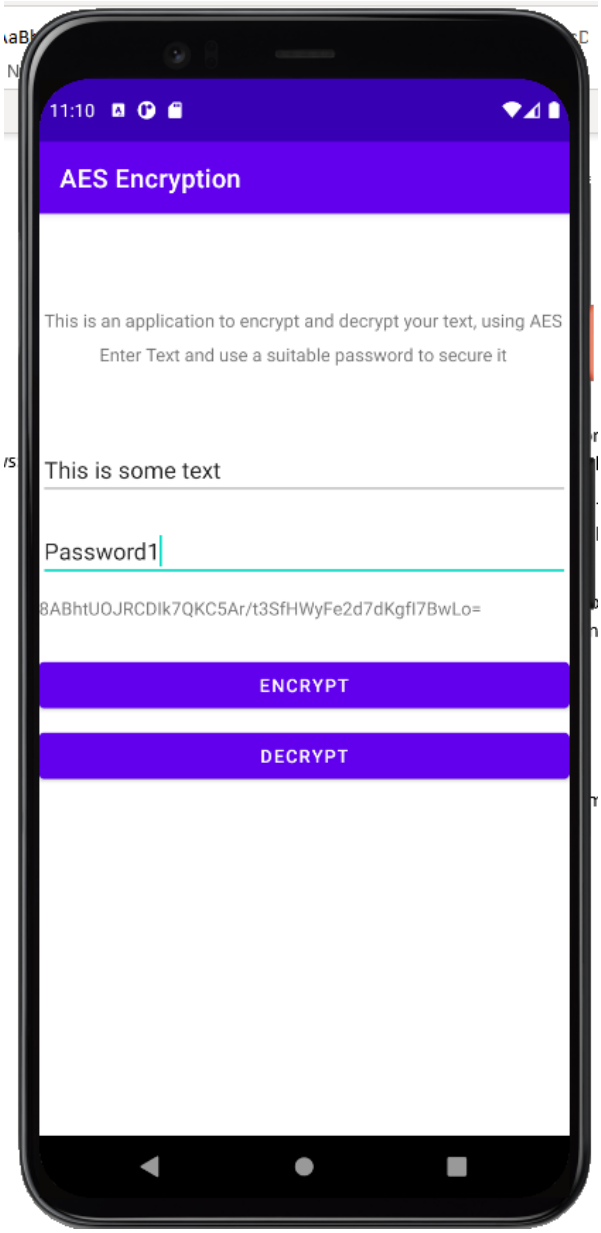

Figure 2: Encrypting text within application

As shown, the encryption result will be displayed beneath the text fields, showing that the text has been encrypted.

Decrypting AES

Now I have shown the encryption method working within the emulator, I will now show the same encrypted text and password being decrypted using my decryption button. This is done by interacting with the decrypt button, this will change the result text field from the encrypted text and change it into the initial text input the user entered, which can be seen demonstrated below:

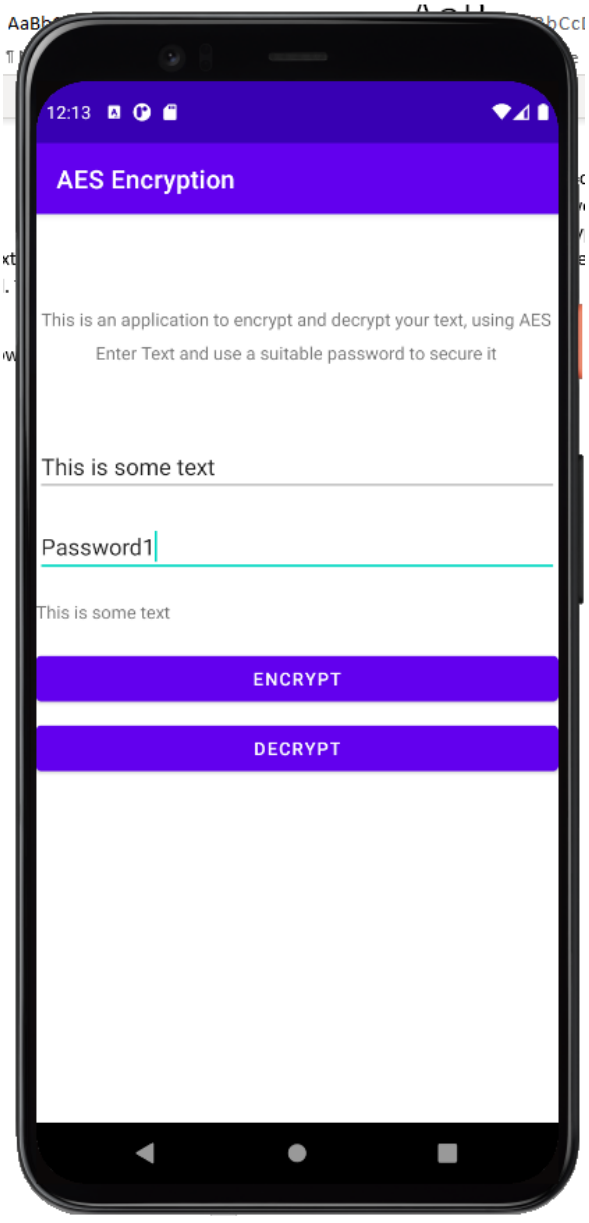

Figure 3: Decrypting text within application

As shown, the encrypted text will be decrypted back into the initial text input.

Entering incorrect password

While decrypting, if the password is changed after the initial encryption, this will display an error message to the user containing, “Incorrect password.” This is done by using Android toast, which is a small message which will be displayed on the screen. This is like a popup notification and alerts the user that they have inputted an incorrect password. This message will only stay on the screen for a short amount of time before disappearing, I felt like this polished up the application.

Here is a demonstration of the user changing the password and the Android toast message appearing on the screen:

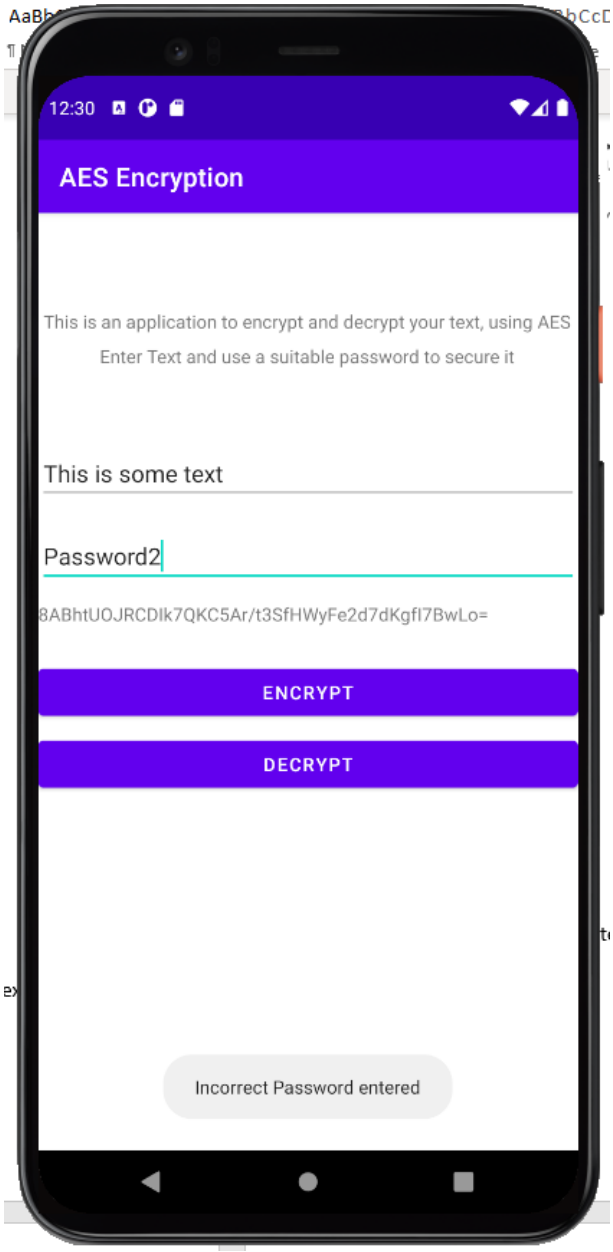

Figure 4: Entering incorrect password

Results and Observations

To conclude on my implementation, I will now discuss my overall results and observations for my implementation. I felt that I accomplished and successfully completed task, to create an AES Encryption application, I also felt as though it relates greatly to my chosen topic, Mobile Device Security. This is due to encryption playing a huge role and popular solution to keeping personal and business data safe. I felt as though I implemented AES well into the application, I felt that AES was the best choice of encryption to implement as it is *"chosen by the U.S. government to protect classified information"* (Cobb, 2020), it is also the most current and popular choice for mobile encryption.

Within my application, the test screenshots prove that the implementation of AES was implemented successfully, to prove this, I will now discuss how I completed this within the code, discussing the main 4 methods I created in the main activity java file.

Encryption Method

Within the encryption, this is where the main encryption processes occur. This is also the method which is called within the buttons and text fields to display the result of the AES encryption. The method declares the data inputted and the password as variables. It then generates a key to be used within the encryption. A cipher is then created to get the instance of AES. Then, the cipher is initialised to encrypt mode. The data is then converted into bytes, and the algorithm begins. The encrypted bytes are then encoded into string and return a new encrypted value.

```
//This is the encryption method
private String encrypt(String Data, String password) throws Exception { //Takes the string and password as variables to complete encryption
            SecretKeySpec key = generateKey(password); //Generates a secret spec key
            Cipher c = Cipher.getInstance(AES); //Creating a cipher to get the instance of AES
            c.init(Cipher.ENCRYPT_MODE,key); //initialise cipher object to encryption mode
            byte[] encVal = c.doFinal(Data.getBytes());
            String encryptedValue = Base64.encodeToString(encVal, Base64.DEFAULT); //Encoding binary values to string
            return encryptedValue; //Returning the encrypted value
 }
```

Figure 5: Code: Encryption method

Decryption Method

The decryption method is where the encrypted text is decoded back into the initial text. It takes the variables string and password from the text field. A key is generated using the password variable, as the specific password inputted by the user is also used in encrypting the text. It then creates a cipher to get the instance of AES. This cipher is then initialised and put into decrypt mode. The binary values are then decoded from bytes back into the string. This is then returned, so therefore the text field can display the outcome.

```
//This is the decryption method
private String decrypt(String outputString, String password) throws Exception{ //Takes the string and password as variables to complete decryption
    SecretKeySpec key = generateKey(password); //Generates a secret spec key
    Cipher c = Cipher.getInstance(AES); //Creating a cipher to get the instance of AES
    c.init(Cipher.DECRYPT_MODE, key); //initialise cipher object to decryption mode
    byte[] decodedValue = Base64.decode(outputString, Base64.DEFAULT); //Decoding binary values to string
    byte[] decValue = c.doFinal(decodedValue);
    String decryptedValue = new String(decValue);
    return decryptedValue; //Returning the decrypted value
}
```

Figure 6 Code: Decryption method

Decode Button

This is a mutator method to interact with the decrypt button once it is clicked on by the user. The try and catch within this method will check the password, which is entered by the user, and matched against the original password to check if it is correct. If the incorrect password has been entered it will display an android toast dialogue box to inform the user that the wrong password has been entered, therefore not decrypting the encrypted message. If the password is correct, then it will decrypt the encrypted text by using the output string and inputted password.

```
//This is the functionality of the decode button
decodeButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) { //When the user interacts with the button it will perform a try and catch to decrypt the encrypted text produced
        try {
            outputString = decrypt(outputString,inputPassword.getText().toString()); //This checks the input password
        } catch (Exception e) {
            Toast.makeText( context: MainActivity.this, text: "Incorrect Password entered", Toast.LENGTH_SHORT).show(); //To display a small message box alerting
            e.printStackTrace(); //For handling exceptions and errors within the try
        }
        outputText.setText(outputString); //This will set the decrypted text back to the initial text in the output text box
    }
});
```

Figure 7: Code: Decode button method

Encode Button

This is also another mutator method with a try and catch, this method will link up the encrypt button to the encrypt method, to display the encrypted text within the text field. It takes the initial text and password input and calls the encrypt method. Then it sets the output text to the output of the encryption, this is followed by a catch to handle any exceptions.

```
//This is the functionality of the encode button
encodeButton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) { //When the user interacts with the button it will perform a try and catch to encrypt the initial text entered
        try {
            outputString = encrypt(inputText.getText().toString(), inputPassword.getText().toString()); //To take the initial text entered and the password input
            outputText.setText(outputString); //This set's the text to the output of the encryption
        } catch (Exception exception) { //Exception handler, to define the code to be executed
            exception.printStackTrace(); //For handling exceptions and errors within the try
        }
    }
});
```

Figure 8: Code: Encode button method

Generating Keys

The method secret key spec's purpose is to generate keys for the encryption and decryption. It uses message direct, which creates a one-way hash. The key size is converted into 256-bits to fulfil the requirement for creating AES with 256-bit keys. The method then completes the digest which performs the AES algorithm on the specified array of bytes, finally creating the new secret key and returning it.

```
//This method is to generate keys
private SecretKeySpec generateKey(String password) throws Exception {
    //Message direct is used to create a one way hash
    final MessageDigest digest = MessageDigest.getInstance("SHA-256"); //Creating the key size of 256-bits
    byte[] bytes = password.getBytes( charsetName: "UTF-8"); //Converting the password into bytes
    digest.update(bytes,  offset: 0, bytes.length); //Updates digest using specified array of bytes
    byte[] key = digest.digest(); //Completing the digest
    SecretKeySpec secretKeySpec = new SecretKeySpec(key,  algorithm: "AES"); //Creating a new key using the AES algorithm
    return secretKeySpec; //Returns the secret key
}
```

Figure 9: Code: Generating code method

Conclusion

To conclude, this coursework has given me a greater insight on the risks that mobile phones face, and most importantly, many different solutions to prevent valuable data being stolen by unauthorised individuals. I have learnt a lot about AES, and understand how to implement this within an application, this will be useful for adding higher levels of security within mobile applications I may develop in the future. I felt as though my implementation went well as I successfully implemented AES and chose the highest key size of 256-bits to maximise security. It has also improved my knowledge of using Java libraries within my code as I have learnt a lot about Java's standard of encrypting data.

If I were to improve my implementation and recreate it, I would like to try to implement a chat system between 2 users on a mobile application, using AES to encrypt messages sent within the chat. I felt as though I could have made my application look more professional by adding a background image, perhaps even experimenting with more features like text to speech (TTS) and having music play in the background.

I felt as though my report went well, I carried out a lot of research and have a greater depth of knowledge about security within mobile devices. However, I feel like I could have covered more examples of real-life scenarios which occurred because of the ongoing threats of mobile device security.

As for the presentation, I really enjoyed the interactive side of this coursework as it allowed me to present my research and hear feedback before the submission. This greatly impacted my implementation, as I originally designed an AES encryptor within NetBeans, this did not relate to mobile devices as greatly as android studio, it shows the encryption happening on an emulated mobile device further strengthening my overall topic. To improve, I would have added more design and spent longer trying to polish it up before the presentations took place.

Overall, I really enjoyed this coursework as it allowed me to complete my own individual research, allowing me to be creative and independent.

References

Buskirk, B. (2019, June 20). *The Advantages And Disadvantages Of Proxy Servers*. Retrieved from Think Computers: https://thinkcomputers.org/the-advantages-and-disadvantages-of-proxy-servers/#:~:text=Proxy%20servers%20can%20be%20configured,%2C%20workplace%2C%20or%20an%20organization.

Cobb, M. (2020, April). *Advanced Encryption Standard (AES)*. Retrieved from SearchSecurity : https://searchsecurity.techtarget.com/definition/Advanced-Encryption-Standard

Dargin, M. (2017, April 3). *Encryption critical for protecting mobile device data*. Retrieved from NetworkWorld: https://www.networkworld.com/article/3187091/encryption-critical-for-protecting-mobile-device-data.html#:~:text=A%20strong%20encryption%20algorithm%20to,the%20few%20unbreakable%20encryption%20algorithms.

fortinet. (2021). *Proxy Server*. Retrieved from Fortinet: https://www.fortinet.com/resources/cyberglossary/proxy-server

Fruhlinger, J. (2020, September 4). *What is phishing? How this cyber attack works and how to prevent it*. Retrieved from CSO Online: https://www.csoonline.com/article/2117843/what-is-phishing-how-this-cyber-attack-works-and-how-to-prevent-it.html

Gontovnikas, M. (2017, December 4). *10 Mobile Security Threats (and What You Can do to Fight Back)*. Retrieved from auth0: https://auth0.com/blog/ten-mobile-security-threats-and-what-you-can-do-to-fight-back/

Infactica. (2020, July 8). *All You Need to Know about Mobile Proxies: How They Work & How to Use Them*. Retrieved from infactica : https://infatica.io/blog/all-you-need-to-know-about-mobile-proxies/

Johansen, A. G. (2020, August 4). *10 benefits of a VPN you might not know about*. Retrieved from Norton: https://us.norton.com/internetsecurity-privacy-benefits-of-vpn.html

Kaspersky. (2021). *Smartphone VPNs: What You Need to Know*. Retrieved from kaspersky: https://www.kaspersky.co.uk/resource-center/threats/why-use-vpn-on-smartphone

Lake, J. (2020, February 17). *What is AES encryption and how does it work?* Retrieved from Comparitech: https://www.comparitech.com/blog/information-security/what-is-aes-encryption/

NordVPN. (2021). *Download NordVPN app for Windows*. Retrieved from NordVPN: https://nordvpn.com/download/

O'Dea, S. (2020, December 18). *Number of mobile devices worldwide 2020-2024*. Retrieved from Statista : https://www.statista.com/statistics/245501/multiple-mobile-device-ownership-worldwide/

RF Wireless World. (n.d.). *Advantages of AES | disadvantages of AES*. Retrieved from rfwirlessworld: https://www.rfwireless-world.com/Terminology/Advantages-and-disadvantages-of-AES.html

Roomi, M. (2020, April 8). *5 Advantages and Disadvantages of Proxy Server | Limitations & Benefits of Proxy Server*. Retrieved from Hitechwhizz: https://www.hitechwhizz.com/2020/04/5-advantages-and-disadvantages-risks-benefits-of-proxy-server.html

Sharma, M. (2020, August 18). *Smartproxy proxy service review*. Retrieved from TechRadar: https://www.techradar.com/uk/reviews/smartproxy

T, N. (2020, May 5). *Advanced Encryption Standard (AES)*. Retrieved from Binary Terms: https://binaryterms.com/advanced-encryption-standard-aes.html

Webopedia Staff. (2021, April 7). *What Is A Proxy Server?* Retrieved from Webopedia: https://www.webopedia.com/reference/proxy-server-types/