Phase I

The Dark Army

September 24, 2016

1 Application Properties

The goal of this application is to provide an end to end encrypted messaging system between users. We will be implementing the application on a mobile device, using android studio as our main IDE. LetsEncrypt is an open certificate authority(c.a) that is free to the public. We will be using the c.a for this project. We will be programming in Java and will also use the Spring framework.The Spring Framework is a Java platform that provides comprehensive infrastructure support for developing Java applications. Spring handles the infrastructure so you can focus on your application.

2 Assets/Stakeholders

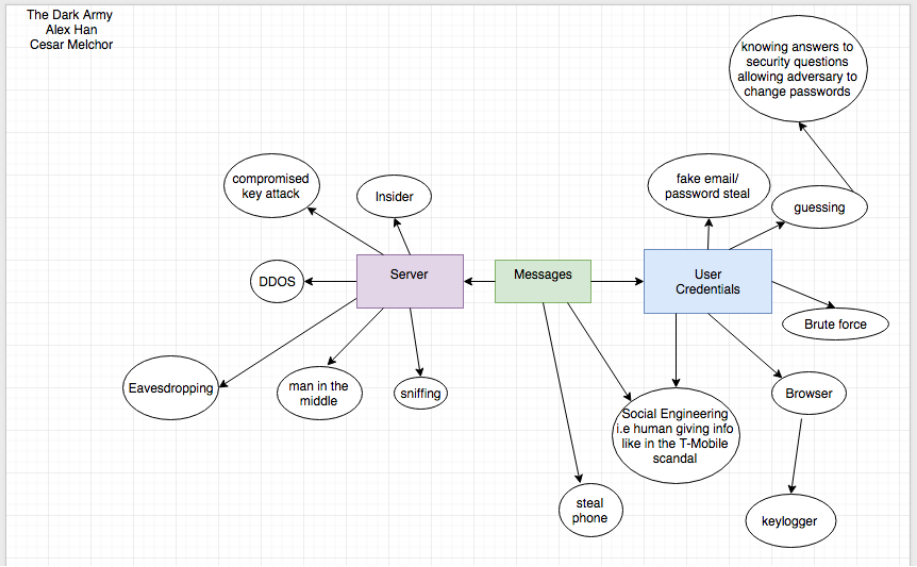
The user's messages, user's log in information, server Stakeholders: We also have to protect the user's log in information. we can use JWT for authentication.

3 Adversarial Model

The messages between the 2 users need to be encrypted so only they can read them. The user credentials need to be authenticated using JWT.

4 Possible vulnerabilities

As we went through the process of finding all possible attack surfaces we came a cross the three most vulnerable points. The first and possibly the most important would be an attack on the Server. We were able to identify multiple entry points. For the server an attack could come possibly come from an inside worker of the sever collecting data for malicious purposes. Other attacks include the infamous DDoS attacks and the man in the middle attack. Eavesdropping is also another surface of attack. The messages themselves are also a vulnerable point. If a person were to steal the device the messages could be compromised. User credentials is another aspect that can be exploited to gain access into the users phone and steal the information directly. Here the adversary could take advantage that the user has a horrible password. Guessing, brute force, and even fake email password steals would allow the adversary to gain access to the account. Bruteforcing/guessing the answers to the user's security questions.



5 Possible related previous work

We looked up Whatsapp and read on how they encrypted their messages and what they did with the keys.

6 Complete description of your solution

We will have a log in in for the users. In order to authenticate the user first we will use JWT and the client will ask for a token from the server. The server will then check the credentials to make sure they are who they are and allow them to proceed or not. From now on, if the client wants to do something like send a message then the server has to verify that the client has the same token. The connection between the server and the client will use SSL so the token can't be taken or seen from any outsiders and it can be secure. In order to distribute public keys we will also use SSL. The message will be encrypted through the Propagating Cipher Block Chaining (PCBC) method. The PCBC mode here is modified version of original Cipher Block Chaining(CBC) mode. With CBC the plain text will split into cipher text blocks of 128 bits each. Each cipher text block is xORed with the previous block and repeated until finished encrypting. The PCBC mode does the same operation but instead of just xORing the cipher text, both cipher text and plain text are xORed.

7 Full Rigorous Analysis

Confidentiality: The whole purpose of this application is to keep all information sent confidential between the two users.
Integrity: This involves maintaining the consistency, accuracy, and trustworthiness of the data within our application as it progresses through time.
Vulnerability: This type of application posses a real threat to those who would

use the product, sending messages back and forth.

8 References

https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation
https://jwt.io/introduction/
http://whatis.techtarget.com/definition/Confidentiality-integrity-and-availability-CIA