REEV.US

An Android Application

ABSTRACT:

REEVUS is an android application that is based on the class website REEV.US/MSCS630S17. It includes all the information that the website has and additional features like log-in and FAQ section. This FAQ section uses encryption to encrypt the questions and answers. REEVUS implements authentication while logging into the app and encryption in the chat area. This application also implements different types of ciphers and helps users understand cryptography in a better way.

Students will have to sign up using their cwid. Only students who registered for the course will be able to sign up successfully and use the application.

REEVUS is an effort to apply cryptography to an application that helps users understand the subject in detail and creating a fun filled learning experience for the users.

INTRODUCTION:

Reev.us gives everyone access to the start page but only registered users can have complete access to the schedule and FAQ section. Users registered for the course will be given a code using which they will be able to register into the app successfully. Reev.us contains all the information about the course including the course objective, textbooks, policies, about the professor etc. There is a schedule section that contains information about labs and assignments that are used for entire semester.

The FAQ section lets students ask any course related question. These questions are encrypted using AES encryption algorithm. Other students can decrypt these

questions by clicking on a button. This will decrypt the question and displays it in normal text form. They can then reply to that question. This answer is again encrypted.

The login section of the application uses MD5 to encrypt the user credentials. These passwords are encrypted and stored in the database.

METHODOLOGY:

Securing the Internet presents great challenges. Applications such as Internet voting, universally available medical records, e-commerce, bank details are all being hindered because of security and privacy concerns. As many cases have proven, user and company data is increasingly being targeted by hackers and cybercriminals resulting in data breaches and targeted attacks. This reason alone should serve as enough warning to those who haven't considered protecting their communications via encryption. Addressing security and privacy concerns requires a combination of technical, social and legal approaches. While encryption doesn't magically convey security, it can still be used to protect a user's identity and privacy. Encryption is a system of mathematical algorithms that encodes user data so that only the intended recipient can read it. It enhances the security of a message or file by scrambling the content. You need the right key to encrypt or decrypt the text. Sender and Recipient have this key that they use to decipher data. There are two types of encryption- symmetric and asymmetric. In symmetric encryption, the sender and the recipient hold the same keys to encrypt and decrypt a message whereas in an asymmetric encryption, a public key is used for encrypting and a private key is used for decrypting it. This application implements symmetric encryption.

Reev.us uses AES 128 bits algorithm to encrypt and decrypt the questions. This application uses a java library to decrypt the messages. Javax.crypto.Cipher package lets us decrypt the text using AES 128 bits. SecretKeySpec class

generates keys that are used for decryption. Cipher class has an init method that takes in 2 parameters, mode and key. Mode specifies whether the text has to be encrypted or decrypted, for example, Cipher.DECRYPT_MODE decrypts the message and Cipher.ENCRYPT_MODE encrypts the message. Dofinal() method decrypts the text and converts it to Base64.

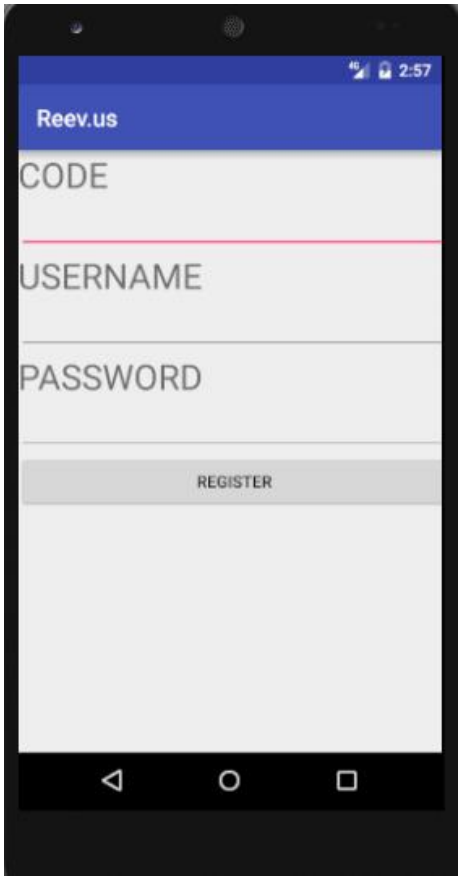
The application uses SHA-1 to encrypt the user credentials. It uses java.security.MessageDigest package to accomplish this. MessageDigest class provides the functionality of a message digest algorithm, such as SHA-1 or SHA-256. These are secure one-way hash functions that take arbitrary-sized data and output a fixed-length hash value.

The text is processed using the update methods. "reset()" can be called to reset the digest. After all the data has been updated, one of the digest methods are called to complete the hash computation. It supports three standard MessageDigest algorithms- MD5, SHA-1, SHA-256.

EXPERIMENTS:

The FAQ section of this application successfully encrypts and decrypts questions using AES 128 bits key. Advanced Encryption Standard (AES) is based on a design principle known as a substitution-permutation network. It has a fixed block size of 128 bits and a key size of 128, 192 or 256. This application uses key size of 128 bits. If a user types in "I have a question" and clicks on "send" button. The question will be displayed as "7989988A8A4C1533756E44075608EC73744D3CCA7A97B8DDB0DF7D1E61B155BA". When the user clicks on "decrypt" button, the encrypted text will be decrypted back to original plain text. The user then types in an answer which is also encrypted and can be decrypted by clicking on the button.

SCREENSHOTS:



This is the "Register" screen that the student uses to register into the app to get complete access to all the sections of the app. A user can successfully register himself to the app only if he enters the correct code. This feature will restrict the application only to those students who are registered for the course.

The screenshot here depicts the FAQ section of the app. The question and answers in this section are encrypted and can be decrypted by clicking on the "decrypt" button.

SYSTEM SPECIFICATIONS:

Hardware Requirements: A mobile phone or a tablet.

Software Requirements: Android OS version 4.3 or above.

Programming Languages: Java, HTML, XML, SQLite

IDE: Android Studio

CONCLUSION:

Reevus is an attempt to apply cryptography to an application that helps users understand the subject in detail and creating a fun filled learning experience for the users. This application also lets students access the course related data with ease without having to browse into the website.

FUTURE ENHANCEMENTS:

I would like to include the Notification feature into the application. This will notify the user every time the schedule is updated or a question is posted. The application may also include few stress-buster games like puzzles, fun quizzes etc that lets users learn the subject through games.

REFERENCES:

- P. (n.d.). ProgrammingKnowledge. Retrieved May 07, 2017, from https://www.youtube.com/channel/UCs6nmQViDpUw0nuIx9c_WvA
- Projects Overview. (n.d.). Retrieved May 07, 2017, from https://developer.android.com/studio/projects/index.html
- Message Digest- Java Platform SE7. (2016, January 11). from https://docs.oracle.com/javase/7/docs/api/java/security/MessageDigest.html
- Tyson, J. (2001, April 06). How Encryption Works. Retrieved May 07, 2017, from http://computer.howstuffworks.com/encryption.htm