iSafe

Team
Ariel Amberden
Kurt Schneider

Motivation
As more eCommerce shifts to mobile phones and tablets, it is important to design security solutions with these devices in mind. Secure transactions are needed for bank transactions, non-bank digital transactions (think Bitcoin), online payment, and in-store payment. Some examples of apps that have sprung up to meet the desire for pay by mobile device in physical stores are Google Wallet, Apple Pay, Square, and Paypal. According to J. Gold Associates Research Report, "organizations lost an average of $92.3 million due to fraudulent mobile transactions for a 3% loss of total revenue per year." There is a clear financial and ethical calling for improvements in mobile eCommerce security. Therefore, we propose to implement an eCommerce application that follows the most current mobile security developments.

Description
Mobile transactions ought to be as secure as traditional online purchases from a desktop computer. We have chosen iPhone as the deployment device and Swift as the programming language. We will use an iPhone emulator to demonstrate the application. iSafeCommerce is meant to be a mock PayPal type app that connects users to a mobile-friendly eCommerce transaction server hosted on Dreamhost. The material to protect are user personal data, transactions, and historical purchase data. The security goals of the project are to guarantee confidentiality, integrity, and availability versus:

1. Outside passive adversaries with the capabilities of a high-end PC currently available on the market, such as eavesdroppers on network traffic and fake websites claiming to be ours.

2. Outside active adversaries with the capabilities of a high-end PC currently available on the market, such as denial of service attacks, unauthorized access via stolen login credentials, and SQL injection attacks.

3. Passive inside adversaries accessing data that they are not authorized to see.

4. Active inside adversaries making changes to data or affecting availability that they are not authorized to do.

The following assumptions shall apply to the success of the project:

1. All adversaries' computational power will be limited to the current market's high end personal computer.

2. All adversaries' message knowledge will be limited to cipher text only and known plain text.

3. A successful attack for an eavesdropper requires that sensitive user data or transaction data be retrieved.

4. A successful attack for a fake site attack requires collection of sensitive user data.

5. A successful denial of service attack requires that transactions are halted. A successful defense may include maintaining the site at a reduced capacity until the exact nature of the attack is discovered and normal functioning returned.

6. A successful unauthorized access requires that the system does not detect and rectify changes or fails to notify the user of unauthorized access.

7. A successful SQL injection attack requires that information be retrieved from the database or changes made without system detection and correction/notification of users of data spilled. The system must be able to reproduce what information was compromised to understand the extent of the damage.

8. In order for insider attackers to be considered successful, they must not be identified by the system. If the system can rectify the changes, reproduce what unauthorized accesses were made, and identify the adversary, then the attack is considered unsuccessful.

Confidentiality will be provided by encrypting all sensitive data and transactions and controlling access to data. To thwart eavesdroppers, encryption libraries will be used that implement AES (Advanced Encryption Standard) encryption for all messages between the user and server systems. To thwart

unauthorized access, strong credentials will be required for login by every user. Protection of these credentials requires mandatory password changes at regular intervals without annual repetition and controlled password resets via email notification upon request. Two-factor authentication will be implemented using the device as the physical object in possession of the user. Three incorrect login attempts and any recognized unauthorized access will result in notification of the user and actions to rectify any changes that may have occurred. Users will be automatically logged out if the app closes.

The site credentials will be verified and an SSL secured connection will be established prior to any interaction with user data or performing any transactions. Sessions will require renewal after a time interval to ensure that keys are refreshed.

Users will be required to provide the minimum information required to complete transactions to limit damage by breaches. Payments may be made using traditional credit cards or with a non-bank payment system (likely to implement Apple Pay to complement use with the iPhone). With the goal of mitigating social engineering attacks, a link to important hints to protect the user's online identity will be included in the site menu bar. Emails will be sent to the user with records of purchases and emails of most recent logins, with a link that the user may click to indicate that the login or purchase was fraudulent.

Tasks

Tasks will be evenly shared among the team members in a scrum style system.

1. Research current industry standards for securing server transactions. (Estimated completion 9/30)
2. Research isolating iPhone app processes and securing transactions. (EC 10/8)
3. Implement database for user data and transaction. (EC 10/15)
4. Write server-side code to process transactions, implement SSL and encryption for network traffic, address SQL injection attacks. (EC 10/29)
5. Write iPhone app to connect to server and send new user data and transactions, implement two-factor authentication. (EC 11/12)

6. Implement user login module including strong password creation, scheduled mandatory password changes, email notifications of failed logins, display last login on member homepage, and auto log out. (EC 11/19)

7. Implement server-side handling of client-identified fraudulent activity, suspicious insider activity, and DOS attacks. (EC 11/26)