BiYing Pan

CS475

Assignment 1

S/KEY Design Document

Introduction

S/KEY is a one-time password software system. Each password used in the system is usable only for one authentication and passwords cannot be re-used. The security of S/KEY relies on the difficulty of reversing cryptographic hash functions.

Crypto Technique

A secure cryptographic hash function SHA3-256 is chosen to apply to the S/KEY system. It is built-in python function from python hashlib. The system is built with SHA3-256 function on the client side and applies n-1 iterations of the hash function to it, then send the request to the server. The server applies the hash function to the request. If the result is the same as the value it stored, the client login is authenticated. The function SHA3-256 produces 256 bits digest of a message. It considered one of good cryptographic hash function because the algorithm takes a short digest of the content. If attackers try to change in the original message, must cause a change in the digest, given file must be infeasible for attacker to create a different file with the same digest.

Communication Model and Storage Data File

A textural interface is used for S/Key system communication model because it is a simple implementation yet an effectively testing this cryptographic technique. Hash secret password can be stored on local file system. The model is prompt with a menu for a user to choose an option. The user allows to either register, login or quit.

Register is for new user to register first before log in. Returning user is not allowed to register. During registration the user must provide his/her name and a secret message. The username is saved as a unique file on the user local machine. The secret message will be used to generate S/Key secret password by implementing a secure cryptographic hash function 1024 time. The result is stored on the user's file system. Th user will have 1024 successful logins before the user must re-register. The user must enter a username to login, provide a secret password that stored on his/her data file, send that password to the server to verify the authentication. If hash password is equal to password n-1 then user is authenticated. After login successful, the secret password used cannot be reuse, it must be removed by the user. By letting user to remove the last password already used, the user can manipulate the current process.

Testing Plan

S/KEY initialization

Test Case 1

1. Runn the S/KEY system
2. Choose option 1 to register
3. Enter a username
4. Enter a secret password to generate S/KEY
5. Create keys successful

Result

```
Choose Option:
1. Register
2. Login
3. Quit
1
Enter your name:
b
Enter your secret password to generate S/Key:
12345678
Your keys are created
```

Test Case 2

1. Runn the S/KEY system
2. Choose option 1 to register

3. Enter a username
4. Prompt “User already exits” if user is a returning user

Result

```
Choose Option:
1. Register
2. Login
3. Quit
1
Enter your name:
b
User already exists.

Choose Option:
1. Register
2. Login
3. Quit
```

Login Authentication with S/KEY

Test Case 1

1. Choose option 2 to login the system
2. Enter a username
3. Locate a data file on local machine, where stored the user keys
4. Pick a last password on the file, then copy
5. Enter password by pasting the password
6. Accept the received password, prompt “Login Successful”

Result

```
Choose Option:
1. Register
2. Login
3. Quit
2
Enter a user name:
b
Enter password:
2547474970804f71b7959622f13d362a8e257f521e58145c0d0de851fa9be7b7
Login successful!
```

Test Case 2

1. Choose option 2 to login the system
2. Enter a username
3. Enter password with the password already used
4. Reject the received password, prompt “Invalid Password”

Result

```
Choose Option:
1. Register
2. Login
3. Quit
2
Enter a user name:
b
Enter password:
2547474970804f71b7959622f13d362a8e257f521e58145c0d0de851fa9be7b7
Invalid password!
```

Test Case 3

1. Choose option 2 to login the system
2. Enter a username
3. Locate a data file on local machine, where stored the user keys
4. Pick a random password on the file, then copy
5. Enter password by pasting the password
6. Reject the received password, prompt “Invalid Password”

Result

```
Choose Option:
1. Register
2. Login
3. Quit
2
Enter a user name:
b
Enter password:
0f5e8acdca1760a4a95984e47b0115e32b9e7e9f029172ce501b414637f3a0f8
Invalid password!

Choose Option:
1. Register
2. Login
3. Quit
```

Test Case 4

1. Choose option 2 to login the system
2. Enter a username
3. Prompt “User does not exist” if the user is a new user

Result

```
Choose Option:
1. Register
2. Login
3. Quit
2
Enter a user name:
a
User does not exist.

Choose Option:
1. Register
2. Login
3. Quit
```

Ending S/KEY System

1. Choose option 3 to exit the system
2. Exit the system successful

Result

```
Choose Option:
1. Register
2. Login
3. Quit
3

Process finished with exit code 0
```