Using Hash Functions to Secure Authentication

Ira A. Fulton Schools of Engineering
Arizona State University

Objective

Objective

Identify proper password handling practices

Two Password Handling Schemes

- Naïve
- Naïve Hash Function

Naïve Password Handling

Store user passwords and compare a user password submission to stored version before authentication.

Problem with Naïve Password Handling

Passwords in Database

Name	Password
DarrenT	qwerty
DraganB	NOTSla4e
JeremyL	blockchain4eva
NakulC	codeitup

Problems

- If database is compromised then attacker has passwords
- If database is not compromised then people internal to organization have passwords
- Extra care needs to be taken when broadcasting password to compare it to database
- Basically everything

Hash Functions Help

Naïve Hash Function Application

- When a user enters a password, hash it and send the hash to the server.
- Compare this value to a stored hash.

Problem with Naïve Hash Function

Problem

This set up is subject to what is called a ***rainbow attack***.

Authentication Table

Name	Password
DarrenT	65E84BE…
DraganB	BB102C7…
JeremyL	E28B7C6…
NakulC	81535EB…

Rainbow Table

Rainbow Table

Password	Password Hash
123456	68D969EE…
123456789	15E2B0D…
qwerty	65E84BE…
12345678	EF797C8…
111111	BCB15F8…
1234567890	C775E7B…

Authentication Table

Name	Password
DarrenT	65E84BE…
DraganB	BB102C7…
JeremyL	E28B7C6…
NakulC	81535EB…

Salt

A ***salt*** is data that is seemingly random. It is combined with other information before hashing. That way a salt would need to be known by an attacker in order to preform a rainbow attack.

Basic Secure Setup

- Let *H* be a hash function. We use | to represent concatenation.
- The user submits *H*(Password) to the server. The server then computes *H*(*H*(Password)|Salt) and compares that to a stored value.

Appropriate Use of Salt for Passwords

- Each user has their own salt.
- Salts should allow enough entropy
- Salts provide entropy that humans don’t usually apply to their password creation.

Other Application of Salts

- Compact blocks use an eight byte result for a hash to index transactions.
- The compact block protocol uses a salt to prevent an attacker creating collisions (Birthday Attack) during block propagation.

Examples of Hash Function

Some Hash Functions

- SHA256
- SHA512
- RPEMD-160

Some Key-Derivation Functions

- PBKDF2 (Password Based Key Derivation Function 2)

Some depreciated hash functions

- md5
- SHA1

Hash Functions Used By Bitcoin

- SHA256
- RPEMD-160

Protecting Passwords - Hash Function Choice

The function PBKDF2

- Takes five arguments.
- Two of those arguments tune how difficult this function is to compute.
- You can basically adjust the difficulty of brute forcing this function.