P1L4 Authentication

What is Authentication?

All requests for resources have to be monitored. Every request must be authenticated and authorized to use the resource.

Authentication: Who are you? Prove it.
Authorization: Does this requester have permission to use the resource? On whose behalf is the request being made.

OS(of the TCB) needs to know who makes a request of a protected resource.
Requests come from processes.

Authentication Goals

Availability: when the correct credentials are presented, the resources should be made available to the processor (on behalf of the user).

No false negatives: if a process presents the correct credentials but is denied access, this is called false negative. These should not happen.

No false positives: if a process presents incorrect credentials but is given access, this is called false positive. These should not happen either.

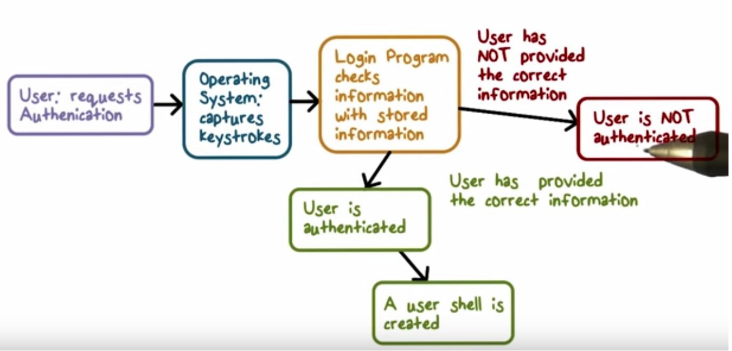
Authentication Quiz

Personal devices are not shared across multiple users, but authentication is still necessary in case of theft. If the device is stolen, authentication will make it unusable to the thief.

How is Authentication Implemented

Three basic methods:

- something only the user knows: such as a password
- something the user has: a token or a smart card.
- something the user is: fingerprint, voice, eye scan, biometric


User: requests Authenication
Operating System: captures keystrokes
Login Program checks information with stored information
User has NOT provided the correct information
User is NOT authenticated
User has provided the correct information
User is authenticated
A user shell is created

1. capture evidence
2. compare it
3. authenticate it

Login Attacks Quiz
False Positive = An attacker correctly guesses Alice's password and logins in as her.
It is positive because an attacker can log in and false because it was not the correct user.

Implementation Quiz
Something you have = A limited lifetime PIN is sent to your smartphone for you to authenticate yourself to the bank.

Threat Modeling of the Password Method
The target: authentication system
Examples of threats:

- Guessing the password
- Impersonating a real login program: a user goes to a website that appears to be the legitimate site for a bank. The user enters his username and password into the site and it is captured by the attacker this is a trojan horse.
- keylogging: grabs your password when you type it in on a computer.

Importance of a Trusted Path
A trusted path = connection between the user and the TCB.
Should be provided by the OS and hardware.

For example: Windows requires CNTL-ALT-DEL will take you to the operating system login software, with no other software in-between.

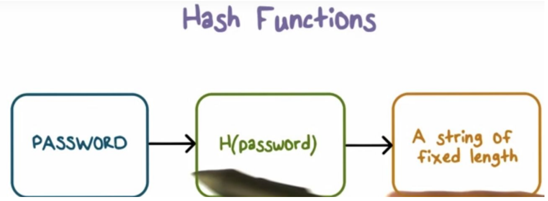
For trusted login path:

- Keyboard and display must have trusted paths to OS
- There can also be a special kind of display under OS control. Possibly a light on the keyboard that turns on when communicating with the keyboard. The challenge to this is do users pay attention to these displays?

Implementing Password Authentication
How to check the password that is supplied.

- Method 1: store a list of passwords, one for each user in the system.
  The file is readable only by the root/admin account.
  Why should the admin know all the passwords.
  Any security breach will result in passwords being known.

- Method 2: do not store passwords, but store something that is derived from them.
  Use a one-way hash function and store the result
  This file should be readable only to the admin account



One way hash function

Hash Functions and Threats

Assume one-way property

If we know common passwords, we can determine their hash.

- Then search the hash file to find those hash values that match, then the hacker will know the password.

Dictionary Attacks

This method requires having access to the hash values and lots of time to test for matches.

- The program has a dictionary of common passwords and try each one (brute force).

Offline Attacks

Take the dictionary of common passwords and compute the hash values for each. Then search the hash file for any matching hashes. Searching for the password can occur offline.

Password Quiz

Without a TCB the user may inadvertently provide his password to a malicious program.

Hashed Passwords Quiz

Shadow password files were added to systems because there is other public information in the /etc/passwd file that is used by various utilities.

Hash Functions Characteristics Quiz

Hash functions used for computing hashed password values should meet the following criteria:

- produced different hashed values for distinct passwords
- its inverse should be very hard to compute

Brute Force Guessing of Passwords

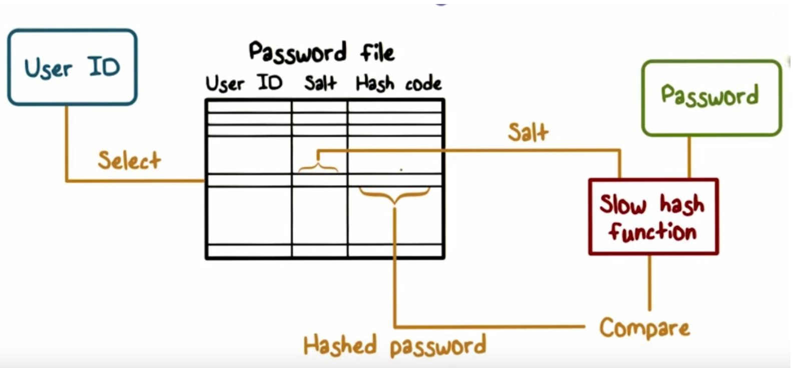
- Hackers can acquire publicly available software can do 10^8 MD5 hashes/sec on a GPU.
- If the password is only six random uppercase, lowercase, and digits then there are 62^6 possible passwords. Using the software, the password can be guessed in about 10 minutes.
- If the password is 8 random characters, it will require about six days to guess the password. If done in parallel this time will be less.

- Passwords are often not really random. Users often choose popular passwords. Hackers can create a rainbow table to break passwords. A rainbow table consists of common passwords and their hashed value.

What if two users pick the same password?

- A random number is added to the password, which will make the hash values different. This means you will have to store this salt with the hash values.

The Unix Password FIle



The hash function is slow to make it harder to guess passwords through brute force.

Brute Force Quiz

If a password has six characters and 72 possible characters for each of the six places, there are 72^6 possible passwords.

Touch Screen Password Quiz

Touch screen passwords are not random, users have a bias.

- Users often start in the upper left corner of the device
- Ease of moving from the current to next point introduces bias.

Problem with Passwords

- As password length and complexity increases, usability suffers.
- Phishing and social engineering take advantage of the fact that users do not often authenticate who is asking for their password.
- Once a password is stolen it can be used many times.
- Humans have a hard time remembering lots of passwords. Usable passwords are easy to guess.

Sys Administrators:

- Never store passwords in the clear
- Only store hashed values and use a random salt
- Avoid general purpose fast hash functions

Users:

- Use a password manager

Other Authentication Methods

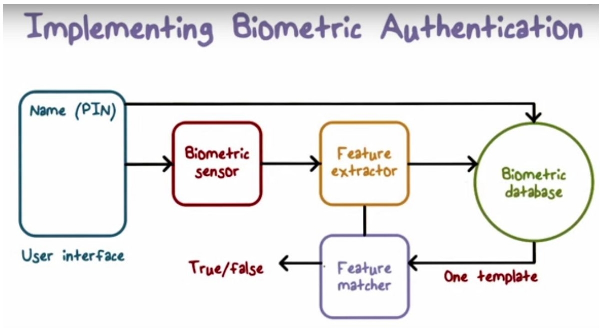
-Something you have (Tokens, smart cards)

- -User must have it
- -May require additional hardware or additional step to authentication
- Implementation of smart cards:
  - -Challenge/Response can be used with smart cards
  - -Increased cost
  - -Attacks can still occur resulting a large scale breach

-Something you are:

- -Biometric method
  - Fingerprints
  - keystroke dynamics
  - voice
  - retina scans
- -Must get same biometric measurement each time
- -Should not get false positives and negatives

The Implementation of Biometric Authentication



Other Authentication Methods

Multi-Factor Authentication

This method will reduce the number of false positives.

The three factors:

- Something you know
- Something you have
- Something you are

These three can be combined into multi-factor authentication.

For example: a password and a pin sent to your phone

ATM card and pin is a two factor authentication

To compromise the authentication, the attacker must defeat both factors.

Authentication over a Network

Remote services require authentication over a network.
This means there is NOT a trusted path.
This introduces new problems into Network Authentication.

- Crypto is needed to secure the network communication

Authentication relies on a Trusted path

Chip and Pin Authentication Quiz
Chip and pin authentication have vulnerabilities - especially to ‘pre-play attacks’.

Biometric Authentication Quiz
Voice recognition can be attacked - through recording the voice or voice synthesis.