User Authentication 1

System Security (CM0625, CM0631) 2023-24
Università Ca' Foscari Venezia

Riccardo Focardi
www.unive.it/data/persone/5590470
secgroup.dais.unive.it

Università Ca'Foscari Venezia

Introduction

Identification is the task of correctly identifying a user or entity

It is typically **required** for enforcing other security properties

Any time the **access to a resource** needs to be regulated, some form of identification is necessary

Examples:

- Users identify into a system when they **login**
- Users identify to mobile network providers through the **SIM card**
- Users identify to the SIM card through a **PIN**
- Users identify to **ATMs** with cards and PINs

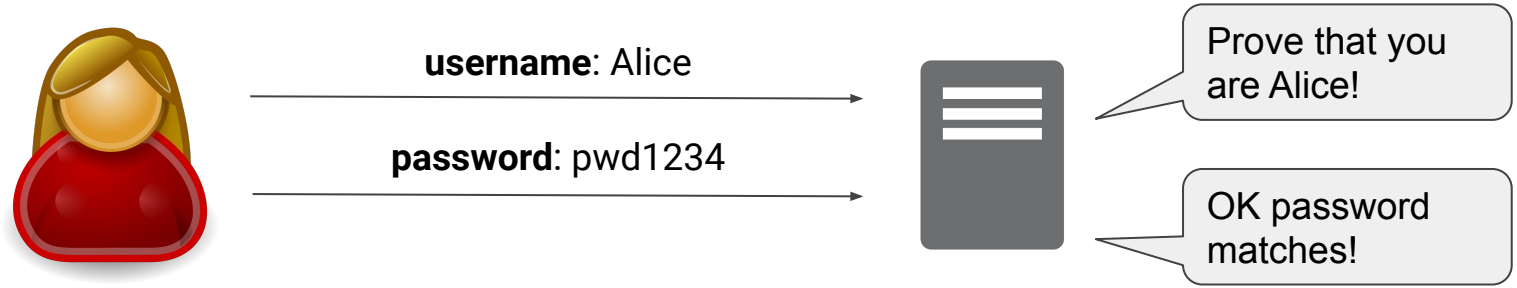
Identification == entity authentication

Identification can be though as **authenticating a user** or, more generally, an **entity**

- Allow a **verifier** to check **claimant's** identity

Example: login-password scheme

- The user **claims** her identity by inserting the **username**
- The system **verifies** the identity by asking for a **secret password**



Properties

An identification scheme should always prevent:

Impersonation, even observing previous identifications

Uncontrolled transferability: the verifier should not **reuse** a previous identification to impersonate the claimant with a different verifier, unless **authorized**

- The verifier has more information available than an attacker, e.g., when the communication is encrypted
- **Example**: same password for different web sites!

Classes of identification schemes

Something known. Check the **knowledge** of a secret

- passwords, passphrases, Personal Identification Numbers (PINs), cryptographic keys

Something possessed. Check the **possession** of a device

- ATM cards, credit cards, smartcards, One Time Password (OTP) generators, USB crypto-tokens

Something inherent. Check **biometric** features of users

- Paper signatures, fingerprints, voice and face recognition, retinal patterns

Passwords

The identity claimed through the **login** information is checked by asking for a corresponding **secret password**

Problem 1: What if the password is *sniffed*?

⇒ stolen passwords allow for **impersonation**
(*weak authentication*: secret is exhibited)

Problem 2: What if password is *guessed*?

⇒ guessed passwords allow for **impersonation**

Problem 3: How are password **stored** on the server?

⇒ an attacker getting into the server might steal all the passwords (might be reused for other servers)

Preventing leakage and guess

Problem 1: What if the password is *sniffed*?

Solution: only use password over encrypted channels

Example 1: passwords and card numbers sent over **`https`**

Example 2: `telnet` was an **insecure** remote terminal client sending passwords in the clear

Problem 2: What if password is *guessed*?

Solution 1: Disable the service after MAX attempts

Example: lock SIM after 3 attempts

Solution 2: Use strong passwords

⇒ useful in offline attacks when the service cannot be disabled