Identification vs. Authentication

Identification
Associating an identity with an individual, process, or request

Authentication
– Verifying a claimed identity

Basis for Authentication

Ideally
Who you are

Practically
Something you know
Something you have
Something about you
(Sometimes mistakenly called things you are)

Something you know

Password or Algorithm
  e.g. encryption key derived from password
Issues
  Someone else may learn it
  Find it, sniff it, trick you into providing it
  Other party must know how to check
You must remember it
How stored and checked by verifier

Examples of Password Systems

Verifier knows password
Encrypted Password
One way encryption
Third Party Validation

Attacks on Password

Brute force
Dictionary
Pre-computed Dictionary
Guessing
Finding elsewhere
### Something you Have

**Cards**
- Mag stripe (≡ password)
- Smart card, USB key
- Time varying password

**Issues**
- How to validate
- How to read (i.e. infrastructure)

### Something about you

**Biometrics**
- Measures some physical attribute
  - Iris scan
  - Fingerprint
  - Picture
  - Voice

**Issues**
- How to prevent spoofing
  - Suited when biometric device is trusted, not suited otherwise

### Other forms of authentication

**IP Address**
**Caller ID (or call back)**
**Past transaction information**
  
  (second example of something you know)

### “Enrollment”

**How to initially exchange the secret.**
- In person enrollment
- Information known in advance
- Third party verification
- Mail or email verification

### Multi-factor authentication

**Require at least two of the classes above.**
- e.g. Smart card plus PIN
- Biometric and password

**Issues**
- Better than one factor
- Be careful about how the second factor is validated. E.g. on card, or on remote system.

### General Problems with Password

**Space from which passwords Chosen**
**Too many passwords**
**And what it leads to**


**Single Sign On**

“Users should log in once
And have access to everything”
Many systems store password lists
Which are easily stolen
Better is encryption based credentials
Usable with multiple verifiers
Interoperability is complicating factor.

**Encryption Based Authentication**

- Proving knowledge of encryption key
  - Nonce = Non repeating value

\[(\text{Nonce or timestamp})K_c\]

**Authentication w/ Conventional Crypto**

- Kerberos or Needham Schroeder

**Authentication w/ PK Crypto**

- Based on public key certificates

**Kerberos**

Third-party authentication service
- Distributes session keys for authentication, confidentiality, and integrity

**Lecture ended Here**

- Remaining slides were covered
  in lecture 7.
**Public Key Cryptography (revised)**

- Key Distribution
  - Confidentiality not needed for public key
  - Solves n² problem
- Performance
  - Slower than conventional cryptography
  - Implementations use for key distribution, then use conventional crypto for data encryption
- Trusted third party still needed
  - To certify public key
  - To manage revocation
  - In some cases, third party may be off-line

---

**Certificate-Based Authentication**

Certification authorities issue signed certificates

- Banks, companies, & organizations like Verisign act as CA’s
- Certificates bind a public key to the name of a user
- Public key of CA certified by higher-level CA’s
- Root CA public keys configured in browsers & other software
- Certificates provide key distribution

---

**Certificate-Based Authentication (2)**

Authentication steps

- Verifier provides nonce, or a timestamp is used instead.
- Principal selects session key and sends it to verifier with nonce, encrypted with principal’s private key and verifier’s public key, and possibly with principal’s certificate
- Verifier checks signature on nonce, and validates certificate.

---

**Secure Sockets Layer (and TLS)**

Encryption support provided between browser and web server - below HTTP layer

- Client checks server certificate
- Works as long as client starts with the correct URL
- Key distribution supported through cert steps
- Authentication provided by verify steps

---

**Trust models for certification**

- X.509 Hierarchical
  - Single root (original plan)
  - Multi-root (better accepted)
  - SET has banks as CA’s and common SET root
- PGP Model
  - “Friends and Family approach” - S. Kent
- Other representations for certifications
  - No certificates at all
    - Out of band key distribution
    - SSH