Internet Infrastructure

Prof. dr. ir. André Mariën
Authentication

A short overview of strong authentication
Authentication Principles

• Authentication = proof of identity
• Methods of proof: “factors”
• Knowledge: password, passphrase, PIN
• Key possession: secret key on token or smartcard
• Physical identity: biometrics
• Strength of proof => strength of authentication
Proof by Knowledge

- Simple to use and deploy
- Weak:
  - no guaranteed uniqueness: secret can be intercepted or passed on
  - Typically easy to remember => easy to guess

My password is “Ispy”

Can the client prove he knows the password?

(c) A. Mariën
Proof by key possession

- Stronger:
  - Better uniqueness guarantees
    - Key is not sent over the network
    - Key storage protection: e.g. smartcard
  - Key strength: n-bits; not that easy to guess
- More difficult to use and deploy:
  - Key generation and key storage access
  - More complex (cryptographic) exchange

Can the client prove he has the key?
Proof by physical identity

• In reality:
  – = proof by knowledge (static password)
    • Can be stolen or copied, e.g. wax fingers
    • Can be intercepted on the network
  – Hard to deploy because of specialized readers
• Bottom line: marginal gain for large effort
Proof by multiple factors

• Combination => N-factor authentication
• Typical examples:
  • 2-factor: PIN-code + key on token
  • 3-factor: PIN + Fingerprint to unlock smart-card
• Strength:
  • Number of factors
  • Implementation details, e.g. type key storage, PIN length, key length
Key-based Authentication mechanisms

- Focus on “proof by key possession”
- Symmetric authentication: shared key
  - One-time-passwords: time or sequence-based, e.g. SecurID or Digipass
  - Challenge-response; e.g. SecurID, Digipass, PPP-CHAP
- Asymmetric authentication: public key authentication
  - Authorized key list, e.g. SSH
  - Digital certificates: through certification authority, e.g. X.509
  - Transitive trust: e.g. PGP
- Typically combined with “proof by knowledge” to ‘unlock’ the key for usage
Symmetric / One-time-password

Mechanism:
• shared key on client and server
• same computation is performed on client and server
  – Time-based: based on key and time
  – Sequence-based: based on key and previous computation
• result = unique password each time

Next password for me

Next password for this user

Uid: p2Xe
Pw: s7aB

q8rs
Symmetric / One-time-password

Properties:

• single exchange, like static password authentication
• => easiest to integrate
• weakness:
  – man-in-the-middle attack
  – freshness
Symmetric / Challenge-response

Mechanism:
- Shared key on client and server
- Challenge sent by server
- Same computation on client and server, using key and challenge, result = evidence
- Mutual authentication possible
Symmetric / Challenge-response

• Properties:
• 2-way exchange: harder to integrate, but not that hard
• Cryptographically stronger
• Still man-in-the-middle weakness
Asymmetric – public key authentication

- Public part of keys are exchanged
- Server sends challenge
- Client signs challenge with private key: evidence
- Mutual authentication possible

Use sender private key

Use sender public key

Equal?
Asymmetric – public key authentication

• Properties:
• Even more complex exchange: even harder to integrate
• Validity of public keys must be verified: extra work
  – Out-of-band exchange, e.g. authorized key list
  – Transitive trust, e.g. PGP
  – PKI infrastructure: signed by Certification Authority
• Solves man-in-the-middle problem by external trust
Key Storage

• Problem: secrecy of private key
• Possibilities:
  • storage on file system: protected with passphrase, e.g ‘soft-tokens’
    – No extra hardware or device drivers required
    – Real risk of key-file theft
    – Simple user experience
  • storage on on-line reader/device, but directly accessible by OS, e.g Rainbow Ikey 1000
    – Much like a removable drive
    – Key can be taken off-line
    – Still real risk of key-file theft
    – Device drivers needed
Key storage (continued)

• Storage on on-line reader/device, cryptographic computations performed on the device, e.g. smartcards, some USB tokens
  – Private key never leaves the device
  – Key-theft risk very low
  – Still risk of abuse because on-line: client compromise
  – Device-drivers needed (e.g. PKCS#11)
  – Tamper-resistance / Tamper-evidence: standard FIPS140-2

• Storage on off-line device: traditional tokens, e.g. SecurID, digipass.
  – Private key never leaves device
  – Programmer needed or factory-programmed
  – Off-line, so no exposure to client compromise
  – Only practical for symmetric key authentication
Sessions

• Application flow = multiple requests / sessions
• Strong authentication: only once
• Session identifier: part of enforcement mechanism
  – Client IP address (firewall client authentication)
  – TCP connection (in-stream)
  – None (HTTP basic authentication)
  – Cookie (HTTP)
  – Session key (SSL)
• Bottom line: strength gain is limited by session mechanism!
Authentication Protocols

Protocols available:

• None (PKI)
• LDAP bind
• RADIUS (TACACS+)
• Proprietary SDK, e.g. SecurID
• RADIUS/EAP
• Kerberos
Authentication Products - Vasco

- Vasco Digipass/GO1:
  - Hardware token
  - 6-8 digit passwords
  - DES/3DES sym. key auth.
  - Factory programmed key
  - No on-device PIN protection
  - No PINpad: OTP only
  - Time- or event synchronous operation
  - Lifetime: 5 years
Authentication Products - Vasco

- Vasco Digipass/DP300
- Hardware token
- DES/3DES symmetric key authentication
- Factory programmed key
- 6-8 digit passwords
- Keypad
- On-device PIN
- Challenge-response
  - time- or event synchronous operation
- Lifetime: 7-10 years
Authentication Products – RSA

- RSA SecurID SD600:
- Hardware token in keyfob formfactor
- Proprietary or AES symmetric key authentication
- Factory programmed key
- 6 digit passwords
- No Keypad
- No on-device PIN
- Time synchronous operation
- Lifetime: up to 5 years
Authentication Products – RSA

- RSA SecurID SD200:
- Hardware token in credit-card formfactor
- Proprietary or AES symmetric key authentication
- Factory programmed key
- 6 digit passwords
- No Keypad
- No on-device PIN
- Time synchronous operation
- Lifetime: up to 4 years
Authentication Products - Datakey

- Datakey Smartcard330
- Smartcard
- Reader: serial, USB or PCMCIA
- On-board key-pair generation and other functions
- CIP client software needed
Authentication Products - Datakey

- Datakey – Rainboy iKey 2032
- USB token
- On-board key-pair generation and other functions
- CIP client software license needed
Authentication Products - Aladdin

• Aladdin eToken PRO USB
• USB token
• On-board key-pair generation and other functions
Connection types

**Connected**

- Transfer data automatically
  - Allows for bigger challenges
  - Allows for bigger responses
  - Allows full certificate operations
- Corrupted platform
  - Connection may be abused
- Less control by user:
  - Data transfer is invisible to the user

**Unconnected**

- Data must be read & typed
  - Limits the amount of data transfer
  - Out of reach of attacker
  - User sees all data in transaction
- Increase bandwidth:
  - optical interface
    - Bar code, QR code
  - Sound interface
    - Old style modem “noises”
Set up

• Symmetric key inside device
• Must be synchronized with server
• Distribute key file
  – Protected keys with customer password
  – Imported & protected with install key
• Key file has ID, token has same ID
  – Logical link: Associate token with user
  – Physical link: Distribute token with ID to user
Logistics

• Tokens with build-in key:
  – Personal
  – If lost, user cannot authenticate anymore
  – Back-up system: possible weaknesses

• Card reader
  – Reader is generic (even across organizations)
  – Card is specific & multi-purpose
Soft tokens

• More and more popular, especially on mobile
• Similar to physical tokens (interchangeable)
• Builds on
  – Integrity of platform
    • Only dedicate app can access key
    • Not rooted, not jail broken
  – Personal device
    • Bind to device (node locking)
• Benefits:
  – Price
  – logistics
Roundtrip systems

• Roundtrip with part of the trip via an independent channel
  – Definition/reality of independent channel!

• Examples:
  – Email roundtrip: send email with code, code via web, loop closed
    • Piggybacks on: mailbox security, pre-registered email
  – SMS roundtrip: code via SMS, code back via web, loop closed
    • Piggybacks on: GSM access, pre-registered GSM number
  – Telephone call back: call, answer and type web page code, loop closed
    • Piggybacks on: pre-registered telephone number
Single sign-on
Single sign-on flavors

• Repository integration
• Front door
• Client side
• Centralized authentication service with tokens
Repository integration

– Multiple systems reuse the same data
– Not really single sign-on: integrated management, same UID/password
– Variations
  • Provisioning: one external master
  • Service integration: back-end authentication service
  • Data integration
Front door

• All access passes through same front door

• Properties:
  – Choke point: maximal control in this point
  – single point of failure: risk
  – in-line system: availability and performance concern

• Web reverse proxy is main example
  – Web single sign on most often uses this method
Client side

• Credential vault, single access key
  – Typical example: safe password on computer

• User certificate
  – Certificate is used for all services that support it
  – Certificate itself is independent from service
Centralized authentication service with tokens

• Prime examples:
  – Kerberos
  – Windows integrated authentication

• Exists also in web variants

• Related to “claims”

• Basics
  – Check if user has a “token” from the authentication server
  – If not, delegate authentication to authentication server
    • Result is “token”
    • Retry
Assertions

Common assertions:

• SAML 2.0 assertions
• WS-federation tokens
• XACML assertions
• Oauth tokens
• Kerberos tokens
STS – IdP

• STS = Secure token service
• IdP = Identity provider
• Typical scheme:
  – To get access, present token from right STS
  – Not available?
  – To get token, present proof of identity from right IdP
  – Not available?
  – To get proof of identity, authenticate with right mechanism
• Uses many “redirections”
Very flexible system

• Choice of:
  – Multiple assertion providers
  – Multiple identity providers
  – Multiple authentication servers

• Configurations

• Mix of technologies possible