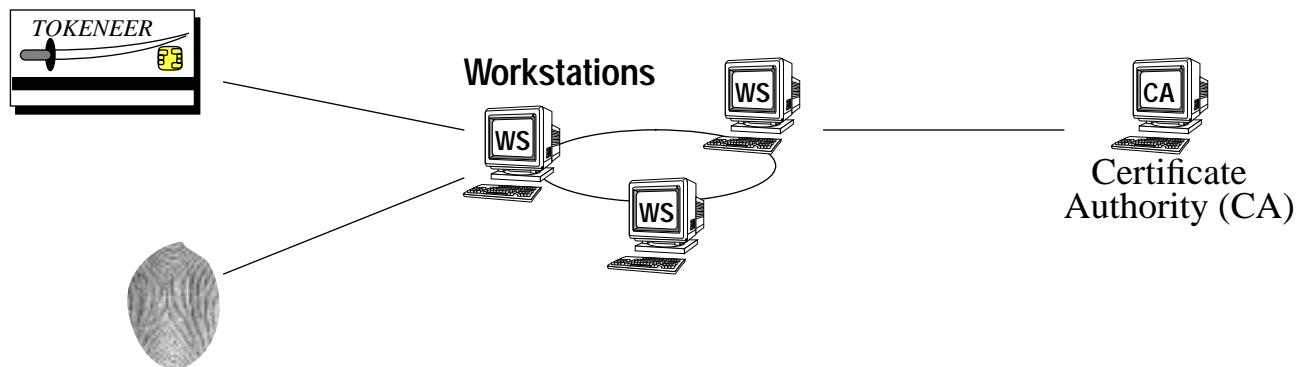


Biometrics, Tokens, & Public Key Certificates

The Merging of Technologies



L. Reinert
S. Luther

R22



Biometrics, Tokens, & Public Key Certificates

The Merging of Technologies

Topics

Authentication Background (Tokens, Public Keys, Certificates, and Biometrics)

Combining Tokens, Public Key, and Certificates

X.509 Certificate Background

Public Key Infrastructure (PKI) Overview

X.509 Attribute Certificates

A Proposed Authentication Information Attribute

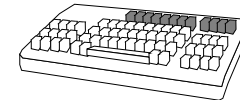
An Example Implementation (Tokeneer)

User Identification Verification Principles

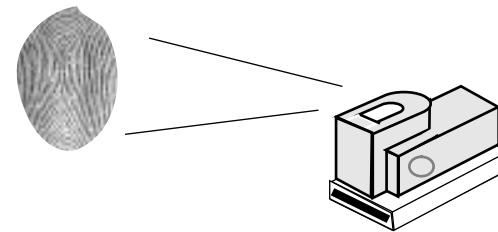
What you have (Tokens)



What you know (passwords, memory phrases, etc.)



What you are (Biometrics)

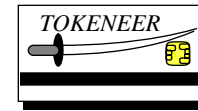


Any data used to support these is considered **Authentication Information (AI)**

How to glue them together?

This is one possible solution, but first some background.....

Tokens (Smartcards)



Small, portable, and potentially cost effective

Typically have a primitive OS which supports a password login feature

Capable of providing Public Key and Symmetric Key Encryption
RSA, Elliptic Curve, DES, etc.

And some support a Hashing function (DES, SHA1, etc.)

Can protect stored data via encryption and reverse engineering techniques

Relatively slow, however improvements will be relatively rapid

Capable of storing (relatively small amounts of) Data

Can support Multiple Applications (via MultOS or Card Java)

Growing Market: Wide variety of applications.

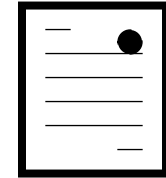
Public Key

- Can be used to provide Confidentiality (via Key Exchange)
- Can be used to support Authentication (via Digital signatures)
- Can be generated on the Token or at a Trusted source (i.e. a CA)
- Secret Component must be held securely by the Entity to which it belongs



RSA is the current standard commercial implementation
KEA/DSA are the government (standards?)
Typically the strength of the algorithm is implied by the bit length associated with the key
(i.e. 1024 bit RSA is harder to break than 512 RSA)
The larger the bit length, the more storage space the key takes, and the longer the processing time.

Certificates



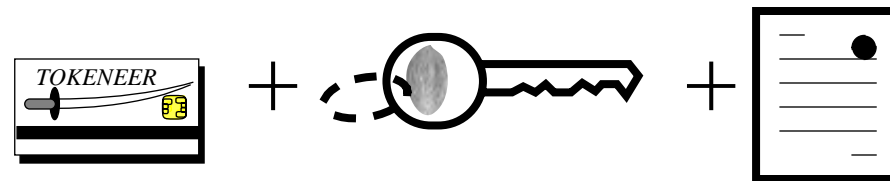
Can be used to provide an authenticated Identity
Requires a Trusted (Certificate) Authority (CA) to sign the certificate

Can be public key or signature certificates
Typically managed by having the CA produce Certificate Revocation Lists (CRLs)

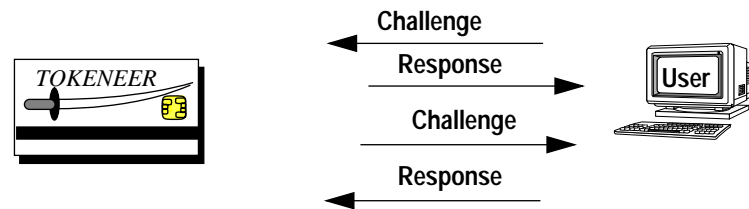
Certificates typically contain the following information:

- Name (identity) of the entity it's associated with
- Issuers name (i.e. the CA)
- Version
- Serial Number
- Validity dates (from.. to..., typically valid one year or more)
- Algorithm identifier
- Public Key Encryption Key or Public Signature Key Data
- CA's Signature

Tokens + Public Key + Certificates



Capable of providing mutual verification per FIPS Pub 196 [8] (i.e. challenge/response)

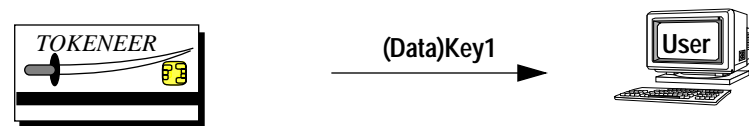


Capable of securely holding User Authentication Information (Passwords, Biometrics, etc.)

(Data)Kn



Capable of securely transferring the Authentication Information to authenticated entities



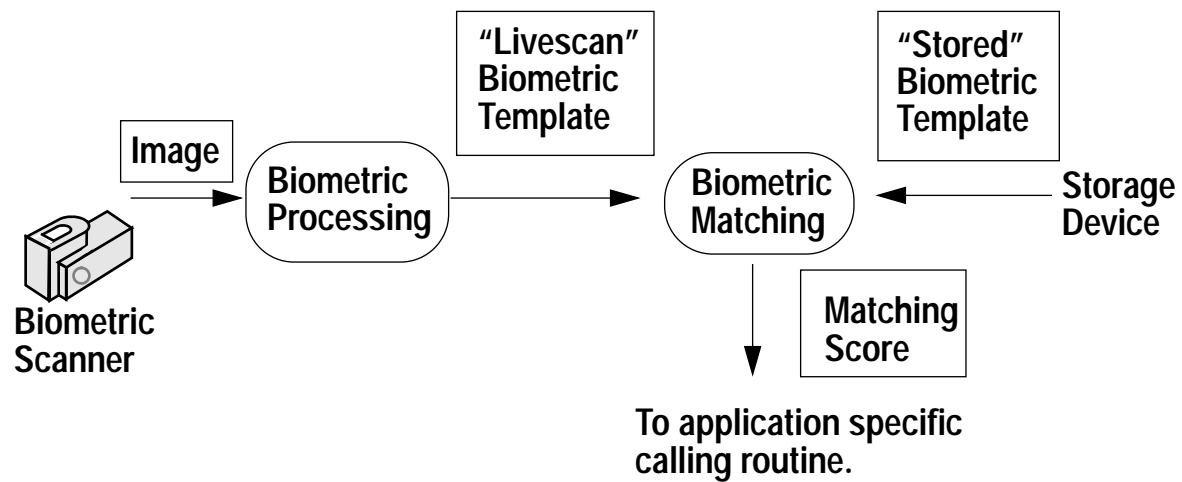
Biometrics

Many different types (Fingerprint, Facial, Retinal, Voice, etc.)

All have pros and cons which fit into differing system requirements.

None are (by themselves) perfect I&A

Most follow this scenario:



Biometrics (Continued)

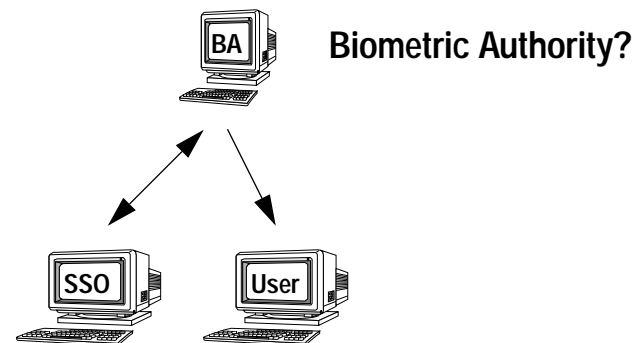
Current systems are typically small, isolated, and proprietary

Work is being done towards commonality

Needs a support infrastructures

Infrastructures required could parallel Public Key Infrastructures (PKI)

But would a Biometric Infrastructures (or AI Infrastructures) Work???



Public Key Infrastructures

Built on a hierarchical “Trust” model

Mature technology

Maturing Business Model (E-Cash/Internet financial transactions)

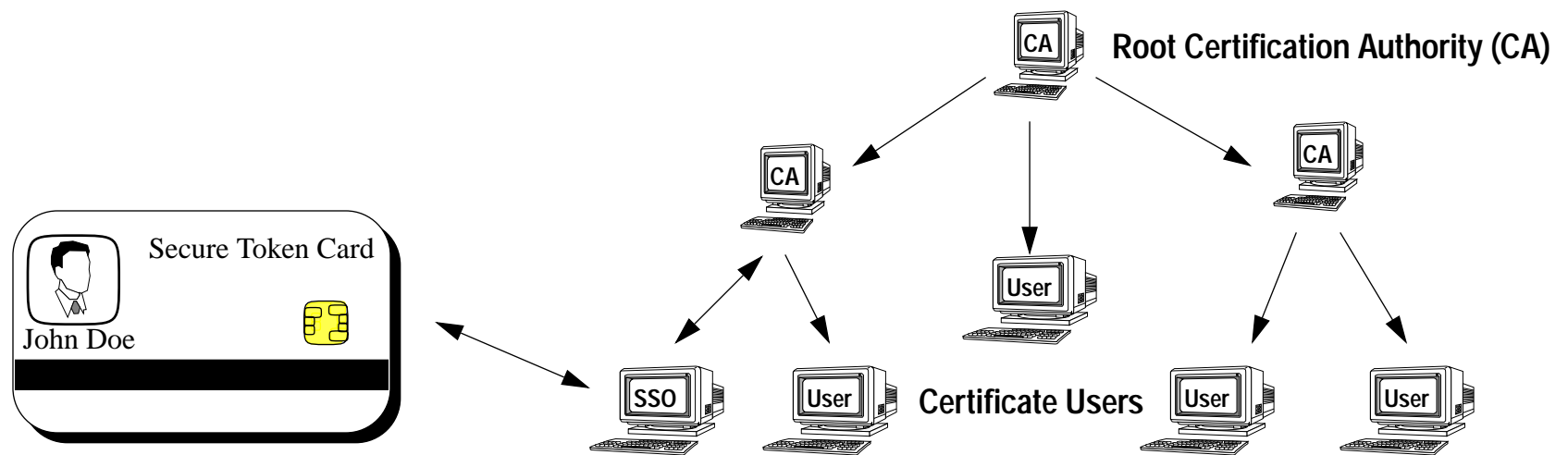
Keys can be generated locally (in the Token) or at the CA.

CA must sign Certificates, Users must verify the CA’s Signature

Revocation Certificates are placed on a Certificate Revocation List (CRL)

Fairly complex key management

Security Policy Management added to handle differing systems



X.509 Public Key Certificates

Of the several types, X.509 is most prevalent in current systems

X.509 is an International Telecommunication Union (ITU) Specification

It is equivalent to ISO/IEC 9598-8 [5], an International Standards Organization Specification

Public Key Certificates are meant to be “public” i.e. not Confidential

Claimed Identity provided by the subject field

Identity and Public key are validated by verifying the CA signature, however....

Identity proven only after successful Challenge/Response

V3 extensions were meant to add addition fields (attributes)

without modifying the base definition [5]:

Authentication Information could go here, but.....

Are you concerned about Confidentiality?

X.509 Public Key Cert

version serialNumber issuer (CA) validity dates subject subjectPublicKeyInfo algorithmIdentifier signatureValue (CA)
--

V3 Extensions

Attributes

Describe a characteristic of the object to which it references [3]

Basic Type (id) and Value construct

Assign a unique identifier

Register with an ISO registered approval authority (ASNI)

X.501 ANS.1 definition[2]

```
AttributeTypeandValue ::= SEQUENCE
```

```
type ATTRIBUTE.&id ({{SupportedAttributes}});
```

```
value ({{ATTRIBUTE.&Type ({{SupportedAttributes}}) { @type }})
```

Some currently defined attributes[4]

Name

Address

Phone

Email address

Company Name

Role

Clearance[5]

What is not clearly defined

Authentication Information (AI), including Biometrics

AI Identifiers

AI Parameters

X.509 Attribute Certificates^[5]

Attribute certificates are used to convey a set of attributes along with a Public Key Certificate identifier or entity name. ^[5]

The attributes were placed in a separate structure to maintain conformance with existing international standards (X.509) ^[5]

Also described in X9.57 an ABA/ANSI specification^[8].

Construct is similar to the X.509 Public Key Certificate, except is specifically set up to hold attributes without the public key

Introduces an Attribute Authority to create/control Attribute Certificates

Attribute Cert

version
subject (baseCertificateID)
issuer (AA)
signature
serialNumber
validity
clearance
role
Authentication Information
algorithmIdentifier
signatureValue (AA)

Authentication Information Attribute for the Attribute Certificate

Authentication Information (AI) properties:

Flexible

Open

Generic

Support different AI's

Support multiple AI's

Support unique parameters

Expandable for future technologies

Support compatibility determination

One Possible Authentication Information Attribute Solution

The AI Attribute should define_[14]:

Authentication Method (ECMA 2.19 _[1])

Passwords

Token

Immutable Characteristics (Biometrics)

Trusted Third Party

Context (Location)

Processing Information

Process Identifier

Version

Parameters

Matching Information

Matching Identifier

Version

Parameters

Authentication Data

Possible Authentication Information ASN.1[2] Description [14]

authenticationInfo ATTRIBUTE ::= { WITH SYNTAX

AuthenticationInfo, ID id-at-TBD }

AuthenticationInfo ::= SEQUENCE {

authenticationMethod [0] AuthenticationMethod, -- defined in ECMA.219

exchangeAI [1] AuthMparm, -- the data, as defined in ECMA.219

biometricInfo BiometricInfo OPTIONAL -- defined in section 5.2.2 of this document }

BiometricInfo ::= SEQUENCE {

processingInfo ProcessingInfo OPTIONAL,

matchingInfo MatchingInfo }

ProcessingInfo ::= SEQUENCE {

processingID OBJECT IDENTIFIER, -- Registered by implementation

processingParmsAuthMparm OPTIONAL, -- Defined in ECMA.219

processingVersionVersion } -- Defined in X.509

MatchingInfo ::= SEQUENCE {

matchingID OBJECT IDENTIFIER, -- Registered by implementation

matchingParmAuthMparm OPTIONAL, -- Defined in ECMA.219

matchingVersionVersion } -- Defined in X.509

--- ECMA.219 definitions

AuthMparm ::= CHOICE {

printableValue [0] Printable String,

integerValue [1] INTEGER,

octetValue [2] OCTET STRING,

bitStringValue [3] BIT STRING,

otherValue [4] ANY } -- defined by authenticationMethod (i.e. the AI)

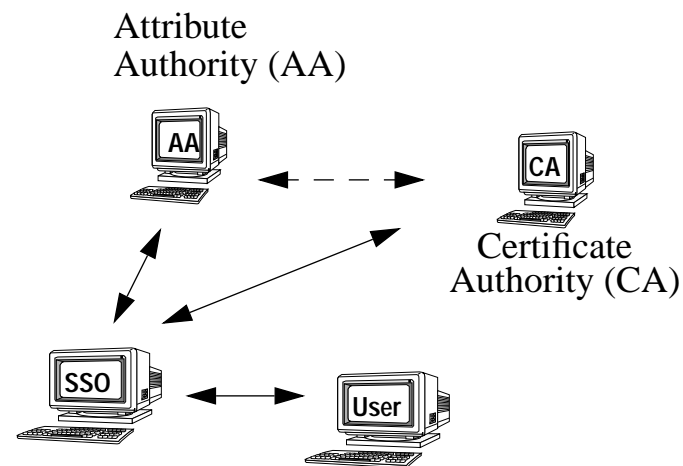
-- X.509 Definitions [5]

Version ::= INTEGER { v1(0) } -- Add versions as needed --

Attribute Authorities^[15]

Attribute Authorities perform similar functions as Certificate Authorities, but specifically are meant to support Attribute Certificates.

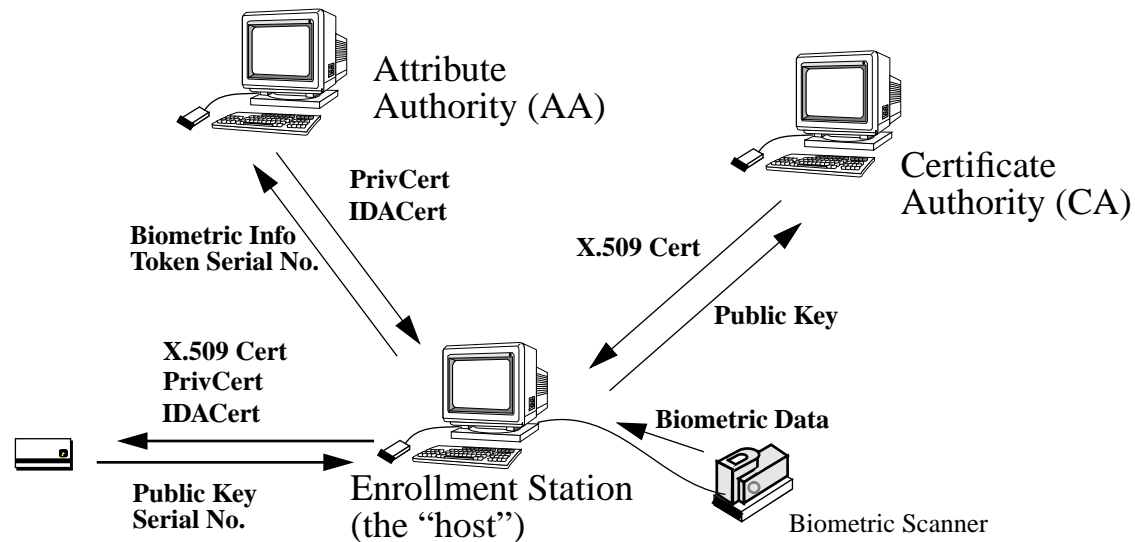
May or may not be the same physical entity as the CA.



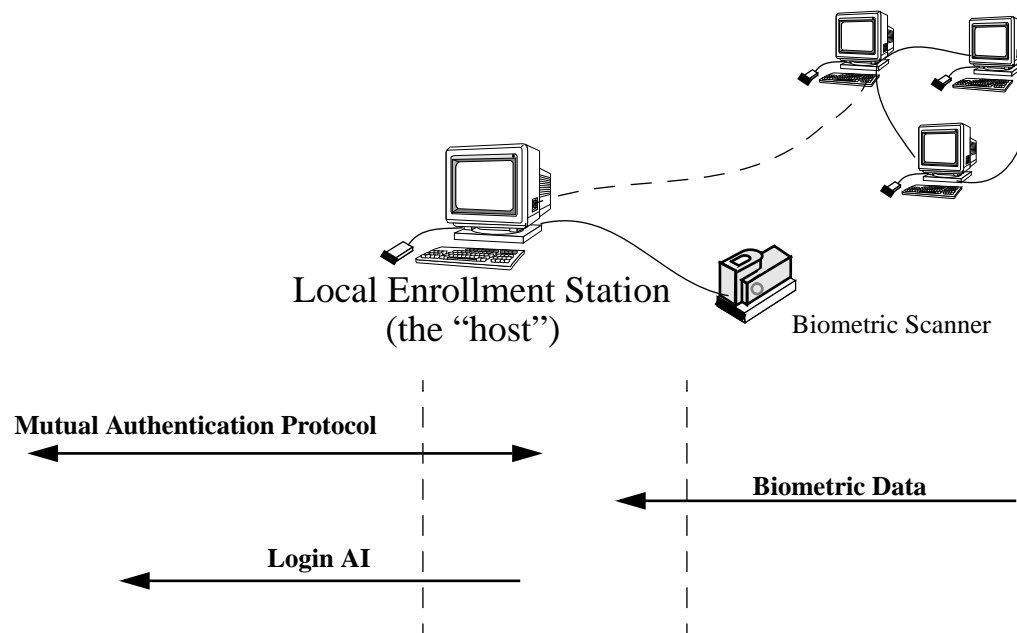
Using Attribute Certificates to provide Confidentiality

A Token Based Scenario (Based upon the Tokeneer system [15])

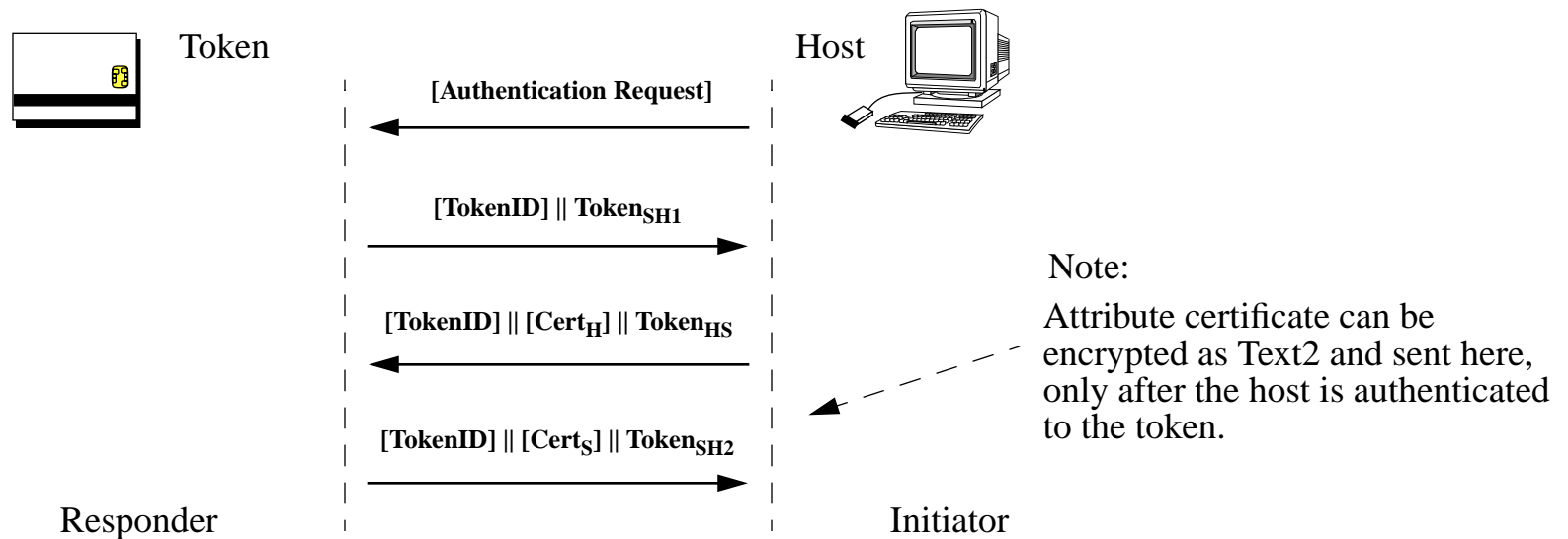
Step1: The System Enrollment Process



Step2: The Local Enrollment Process



FIPS Pub 196_[8] based Mutual Authentication Protocol



Items in [] are optional

Where:

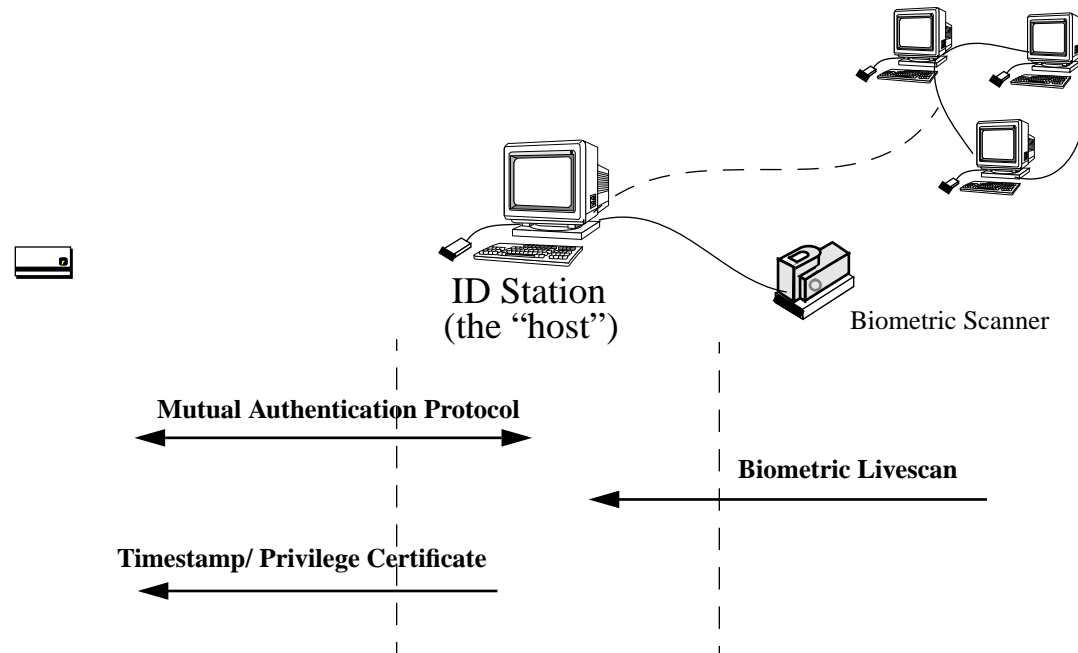
$$\text{Token}_{SH1} = R_S \parallel [\text{Text}_1]$$

$$\text{Token}_{HS} = R_H \parallel [R_S] \parallel [S] \parallel [\text{Text}_3] \parallel s_{SH}(R_H \parallel R_S \parallel [S] \parallel [\text{Text}_2])$$

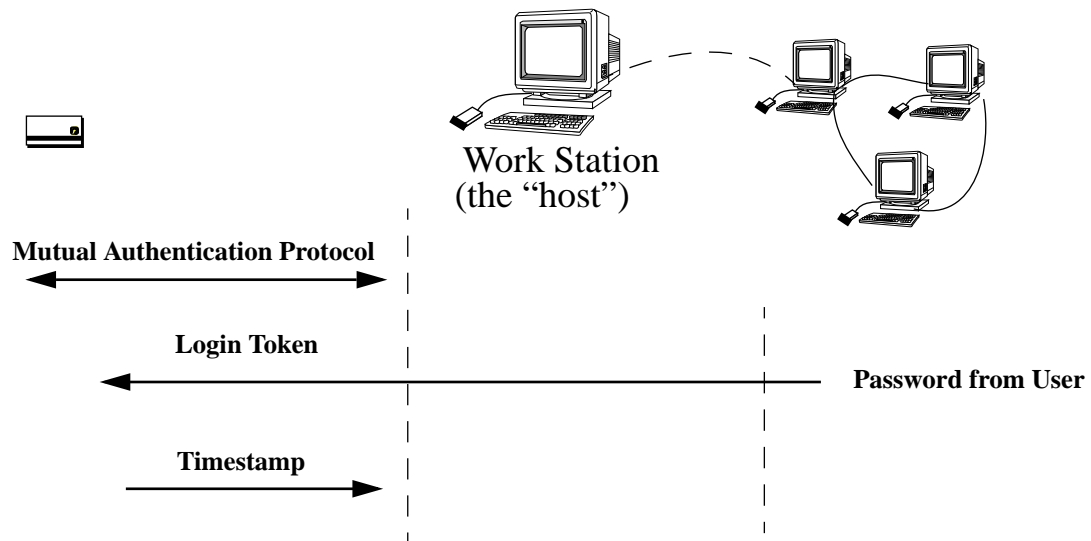
$$\text{Token}_{SH2} = [R_S] \parallel [R_H] \parallel [H] \parallel [\text{Text}_5] \parallel s_{SH}(R_S \parallel R_H \parallel [H] \parallel [\text{Text}_4])$$

Text: Other data to be appended, such as an attribute certificate (e.g. Text 2)

Step3: The Domain Entry: Process User Verification



Step4: User Verification: Workstation Login



Variations on the Token Based Scenario

Trusted Registry

Biometric template response is sent to a central “Trusted Registry”.
Biometric verification performed at the registry.
Challenge/Response could be between token and the Trusted Registry.
Similar to the Trusted Third Party (TTP) scenario.

Local Verification

Attribute certificate is stores in a local database.
Local host verifies the signature on the attribute certificate before verification.
Local Host performs biometric verification and token challenge/response.

Token Verification

Host sends biometric scan to token, token performs biometric verification.
This scenario assumes the token can be trusted more than the host.
Not feasible now, but newer generation of token may be capable.

Issues

Token's cost for large scale deployment

Token processing time (Total processing time should be < 2 seconds)

Token storage (Attribute Certificate could be stored in Host system)

Infrastructure's cost and complexity

Workstation Trust

Consensus on a Authentication Information Attribute

Register the AI Attribute with ANSI

Conclusions

Combining Biometrics, Tokens, and Certificates is possible with today's technology

A new attribute should be created to handle Authentication Information

An Infrastructure must be created to support Authentication Information

The Tokeneer system is a testbed specifically designed to prototype this functionality

Reference Documents

- [1] ECMA. ECMA-219, "Authentication and Privilege Attribute Security Application with related key destruction functions." second edition, 1996.
- [2] ISO/IEC. ISO/IEC 8825, "Information technology - ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)." first edition, 1995-10-15.
- [3] ISO/IEC. ISO/IEC 9594-2, "Information technology - Open Systems Interconnection -The Directory: Models." 1995.
- [4] ISO/IEC. ISO/IEC 9594-6, "Information technology - Open Systems Interconnection -The Directory: Selected attribute types." 1995-09-15.
- [5] ISO/IEC. ISO/IEC 9594-8, "Information technology - Open Systems Interconnection -The Directory: Authentication framework." 06/97.
- [6] ABA, X9.57-199x, "Public Key Cryptography For the Financial Service Industry: Certificate Management", Working draft, June 21,1996.
- [7] NIST. FIPS Pub 190, "Guideline for the use of Advanced Authentication Technology Alternatives." September 28, 1994.
- [8] NIST. FIPS Pub 196, "Entity Authentication Using Public Key Cryptography." February 18, 1997.
- [9] NSA R223. "TOKENNEER Operational Concept Description." Build 1: version 1.0, November 25, 1996.
- [10] NSA R223. "TOKENNEER System/Subsystem Specification (Requirements Document)." Build 1: version 1.0, November 24, 1997.
- [11] NSA R223. "TOKENNEER Authentication Protocol for Smartcards," version 1.0, 1998.
- [12] NCSC. NCSC-TG004-88, "Glossary of Computer Security Terms." 21 October 1988.
- [13] Reinert, L. and S. Luther. "User Authentication Techniques Using Public Key Certificates Part1: Certificate Options." 24 December 1997.
- [14] Reinert, L. and S. Luther. "User Authentication Techniques Using Public Key Certificates Part2: Authentication Information Including Biometrics." 31 December 1997.
- [15] Reinert, L. and S. Luther. "User Authentication Techniques Using Public Key Certificates Part 3: An Example Implementation." 27 February 1998.