INA – Volume 1

Sylvain MARET
Version 1.1 Released

2014-03-07

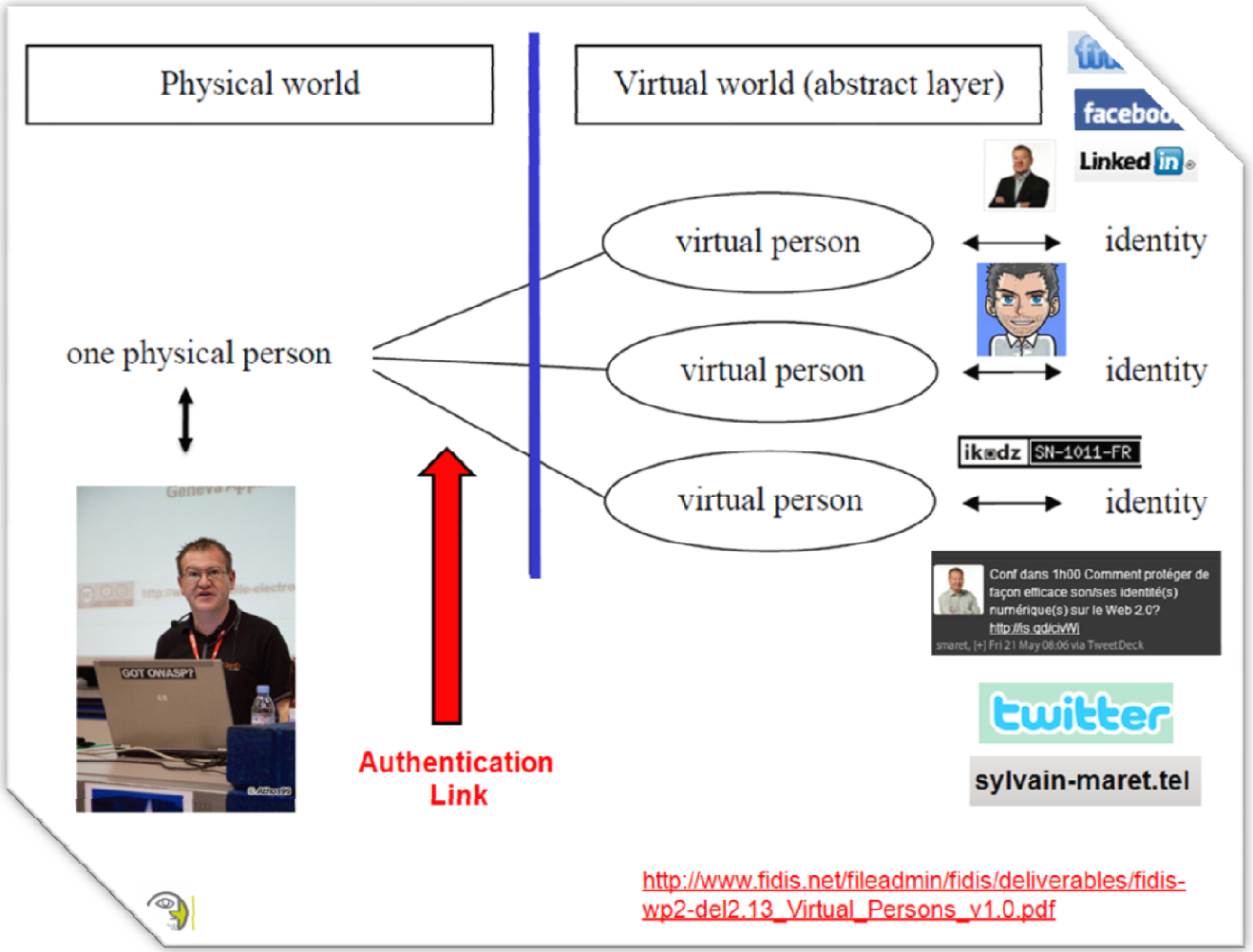
Who am I?

- ICT Security Consultant
  - Principal Cyber Security Architect at Kudelski Security
  - Expert at Engineer School of Yverdon-les-Bains
  - Member of board OpenID Switzerland
  - Co-founder Application Security Forum #ASFWS
  - OWASP Member Switzerland
  - Author of the blog: la Citadelle Electronique
  - http://ch.linkedin.com/in/smaret or @smaret
  - http://www.slideshare.net/smaret
- Chosen field
  - AppSec & Digital Identity Security
  - Cyber Security

Agenda Volume 1

- C0 - Introduction
- C1 - Definition
- C2 - Tokens / Authentication factors
- C3 – Password
- C4 - One Time Password - OTP
- C5 - OTP / OATH standars
- C6 - OTP solution
- C7 - AuthN PKI
- C8 - Biometrics
- C9 - OATH approach

Digital Identity ?



Definition Wikipédia French

Identité numérique

Cette page d'homonymie répertorie les différents sujets et articles partageant un même nom.

L'**identité numérique** peut se référer à :

- l'identité au sens logique (A = A)
- l'identité numérique sur Internet (ou cyberidentité)

Sur les autres projets Wikimedia :

identité numérique, sur le Wiktionnaire

Voir aussi [modifier]

Identité numérique (Internet)

Pour les articles homonymes, voir identité numérique (homonyme).

L'identité numérique peut être définie comme un lien technologique entre une entité réelle (la personne) et une entité virtuelle (sa ou ses représentation(s) numériques).

Definition

Identity

- A set of attributes that uniquely describe a person or information system within a given context.

Source = NIST Special Publication 800-63-1

Authentication

- The process of establishing confidence in the identity of users or information systems.

Source = NIST Special Publication 800-63-1

Electronic Authentication (E-Authentication)

- The process of establishing confidence in user identities electronically presented to an information system.

Source = NIST Special Publication 800-63-1

Claimant

- A party whose identity is to be verified using an authentication protocol.

Source = NIST Special Publication 800-63-1

Subscriber

- A party who has received a credential or token from a CSP.

Source = NIST Special Publication 800-63-1

Token

- Something that the Claimant possesses and controls (typically a cryptographic module or password) that is used to authenticate the Claimant's identity.

Source = NIST Special Publication 800-63-1

TokenCode / PassCode

- TokenCode = OTP Display

- PassCode = PIN Code * TokenCode

Credential

- An object or data structure that authoritatively binds an identity (and optionally, additional attributes) to a token possessed and controlled by a Subscriber.

Source = NIST Special Publication 800-63-1

Identity Proofing

- The process by which a CSP and a Registration Authority (RA) collect and verify information about a person for the purpose of issuing credentials to that person.

Source = NIST Special Publication 800-63-1

Credential Service Provider (CSP)

- A trusted entity that issues or registers Subscriber tokens and issues electronic credentials to Subscribers. The CSP may encompass Registration Authorities (RAs) and Verifiers that it operates. A CSP may be an independent third party, or may issue credentials for its own use.

Source = NIST Special Publication 800-63-1

Registration Authority (RA)

- A trusted entity that establishes and vouches for the identity or attributes of a Subscriber to a CSP. The RA may be an integral part of a CSP, or it may be independent of a CSP, but it has a relationship to the CSP(s).

Source = NIST Special Publication 800-63-1

Verifier

- An entity that verifies the Claimant's identity by verifying the Claimant's possession and control of a token using an authentication protocol. To do this, the Verifier may also need to validate credentials that link the token and identity and check their status.

Source = NIST Special Publication 800-63-1

Relying Party (RP)

- An entity that relies upon the Subscriber's token and credentials or a Verifier's assertion of a Claimant’s identity, typically to process a transaction or grant access to information or a system.

Source = NIST Special Publication 800-63-1

Authentication Protocol

- A defined sequence of messages between a Claimant and a Verifier that demonstrates that the Claimant has possession and control of a valid token to establish his/her identity, and optionally, demonstrates to the Claimant that he or she is communicating with the intended Verifier.

Source = NIST Special Publication 800-63-1

AuthN & AuthZ

- Aka authentication process

- Aka authorization process

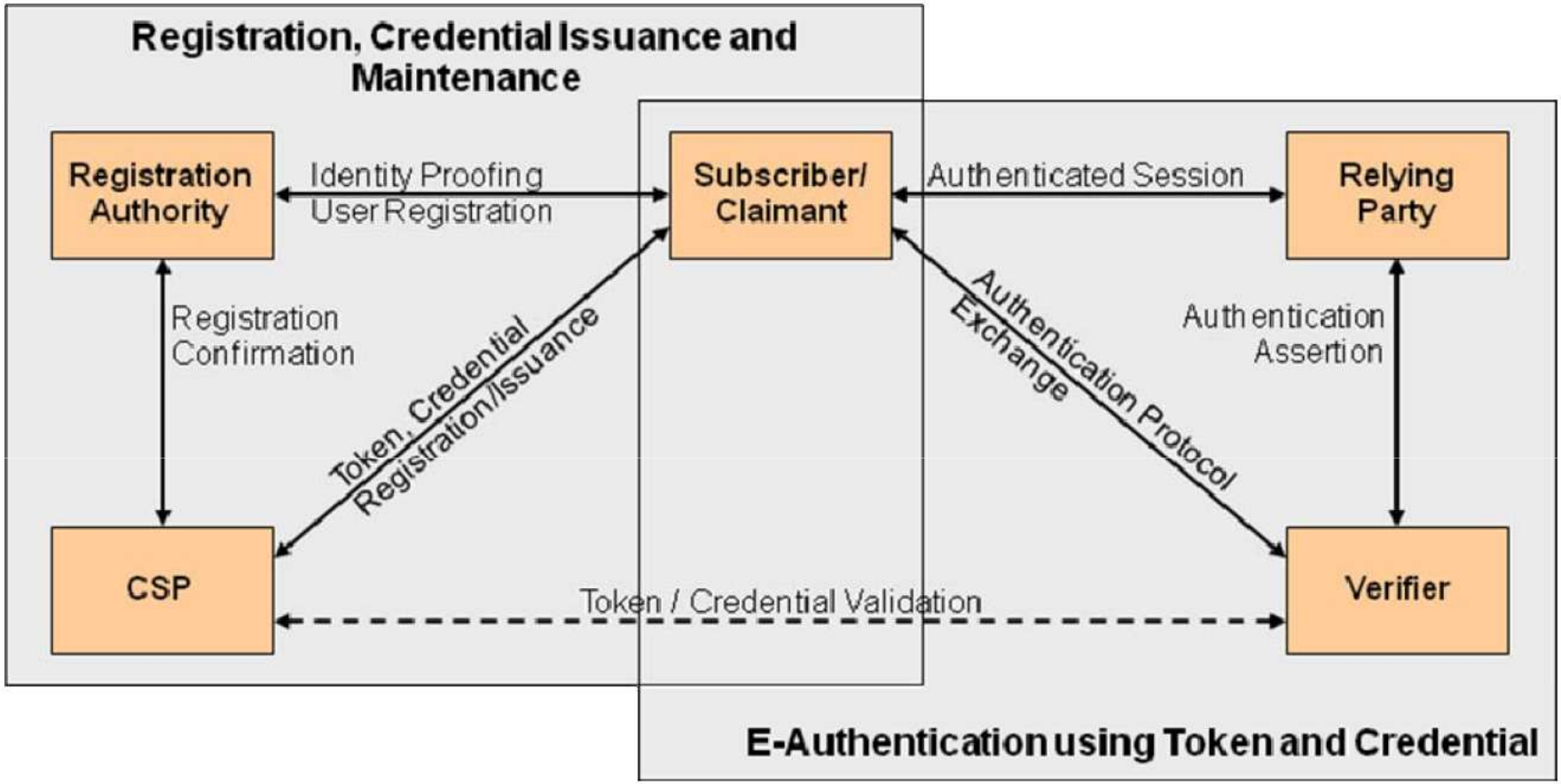
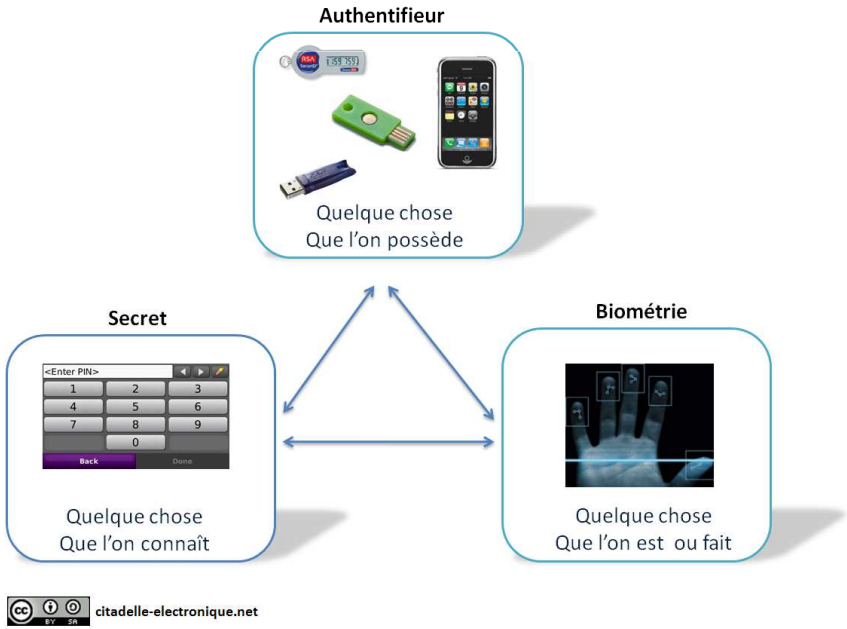


Figure 1 - *The NIST SP 800-63-1 E-Authentication Architectural Model*

Tokens / Authentication factors

Authentication factors

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



Strong Authentication / Multi-factor authentication

- Multi-factor authentication refers to the use of more than one of the factors listed bellow:
  - Something you know
  - Something you have
  - Something you are

Two-factor authentication

- Two-factor authentication
  - TFA
  - T-FA
  - 2FA

Knowledge factors: "something the user knows"

- Password
  - password is a secret word or string of characters that is used for user authentication.
- PIN
  - personal identification number (PIN) is a secret numeric password.
- Pattern
  - Pattern is a sequence of cells in an array that is used for authenticating the users.

Possession factors: "something the user has"

- Tokens with a display
- USB tokens
- Smartphone
- Smartcards
- Wireless (RFID, NFC)
- Etc.

Inherence factors: "something the user is or do"

- Physiological biometric
  - Fingerprint recognition
  - Facial recognition system
  - Iris recognition
  - Etc.
- Behavioral biometrics
  - Keystroke dynamics
  - Speaker recognition
  - Geo Localization
  - Etc.

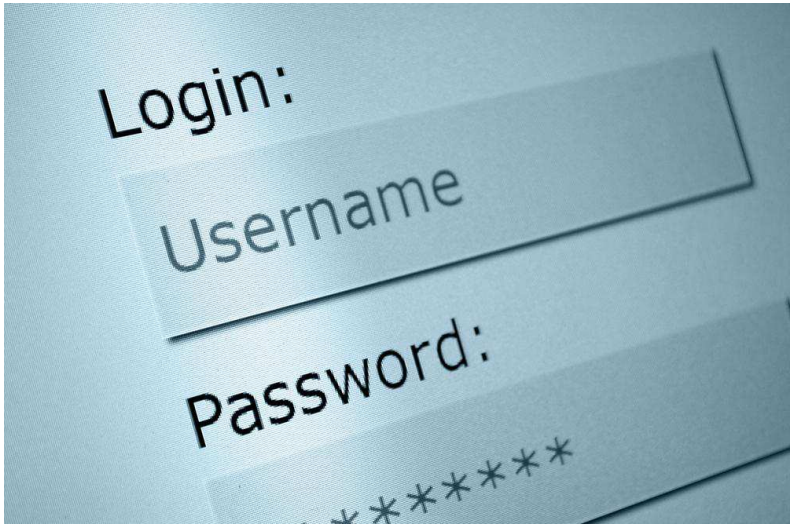
PASSWORD

http://www.wired.co.uk/magazine/archive/2013/01/features/hacked

http://www.wired.com/wiredenterprise/2013/01/google-password/

Password Factor

- Something you know
- PIN Code
- Password
- Passphrase



- Aka 1FA

Password Entropy / Password strength

- Password strength is a measure of the effectiveness of a password in resisting guessing and brute-force attacks.

Password Entropy / Password strength

Entropy per symbol for different symbol sets

Symbol set	Symbol count *N*	Entropy per symbol *H*
Arabic numerals (0–9) (e.g. PIN)	10	3.322 bits
hexadecimal numerals (0–9, A-F) (e.g. WEP keys)	16	4.000 bits
Case insensitive Latin alphabet (a-z or A-Z)	26	4.700 bits
Case insensitive alphanumeric (a-z or A-Z, 0–9)	36	5.170 bits
Case sensitive Latin alphabet (a-z, A-Z)	52	5.700 bits
Case sensitive alphanumeric (a-z, A-Z, 0–9)	62	5.954 bits
All ASCII printable characters	95	6.570 bits
All extended ASCII printable characters	218	7.768 bits
Diceware word list	7776	12.925 bits

http://en.wikipedia.org/wiki/Password_strength

Password Entropy / Password strength

Lengths *L* of truly randomly generated passwords required to achieve desired a password entropy *H* for symbol sets containing *N* symbols.

Desired password entropy *H*	Arabic numerals	Hexadecimal	Case insensitive Latin alphabet	Case insensitive alphanumeric	Case sensitive Latin alphabet	Case sensitive alphanumeric	All ASCII printable characters	All extended ASCII printable characters	Diceware word list
32 bits	10	8	7	7	6	6	5	5	3
40 bits	13	10	9	8	8	7	7	6	4
64 bits	20	16	14	13	12	11	10	9	5
80 bits	25	20	18	16	15	14	13	11	7
96 bits	29	24	21	19	17	17	15	13	8
128 bits	39	32	28	25	23	22	20	17	10
160 bits	49	40	35	31	29	27	25	21	13
192 bits	58	48	41	38	34	33	30	25	15
224 bits	68	56	48	44	40	38	35	29	18
256 bits	78	64	55	50	45	43	39	33	20
384 bits	116	96	82	75	68	65	59	50	30
512 bits	155	128	109	100	90	86	78	66	40
1024 bits	309	256	218	199	180	172	156	132	80

http://en.wikipedia.org/wiki/Password_strength

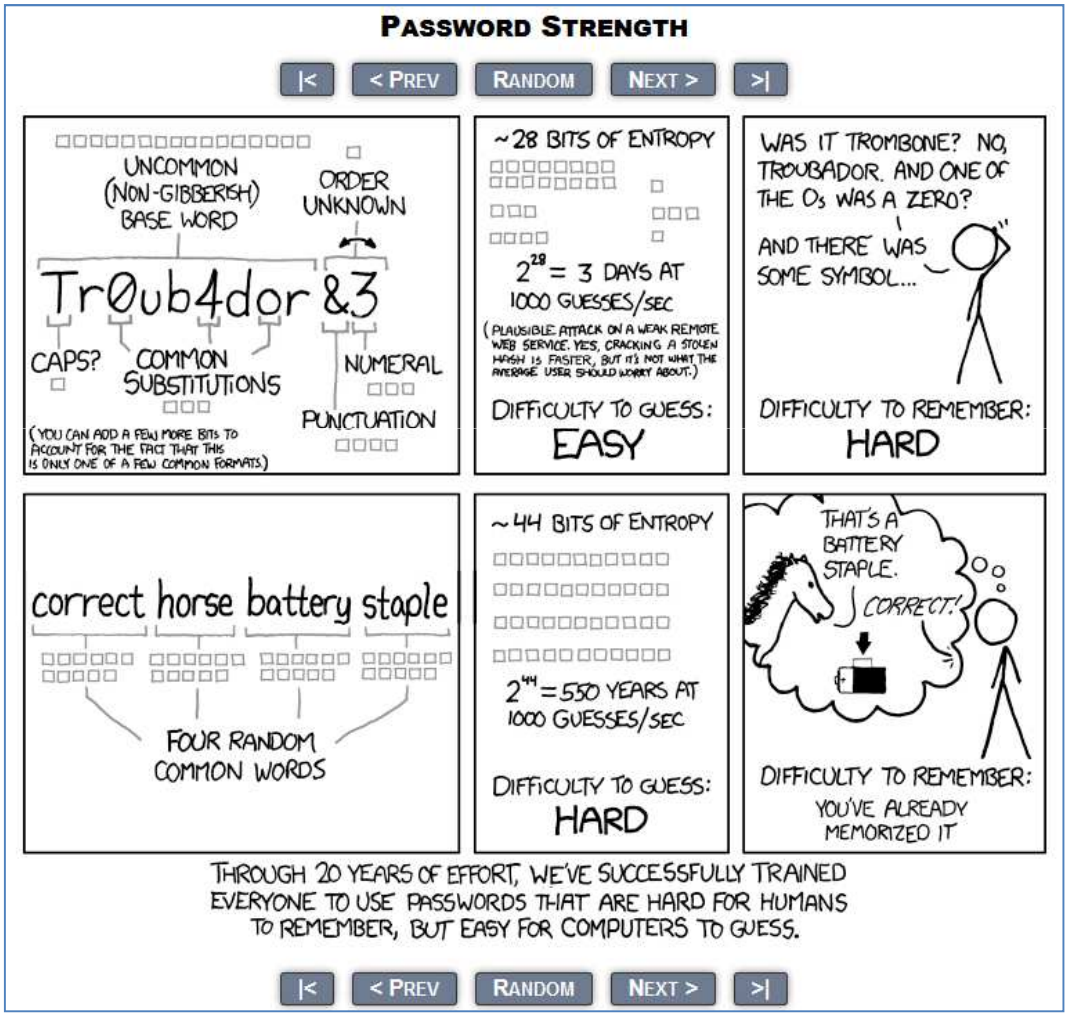
Characteristics of weak passwords

- based on common dictionary words
  - Including dictionary words that have been altered:
    - Reversed (e.g., “terces”)
    - Mixed case (e.g., SeCreT)
    - Character/Symbol replacement (e.g., “$ecret”)
    - Words with vowels removed (e.g., “scrt”)
- based on common names
- short (under 6 characters)
- based on keyboard patterns (e.g., “qwertz”)
- composed of single symbol type (e.g., all characters)

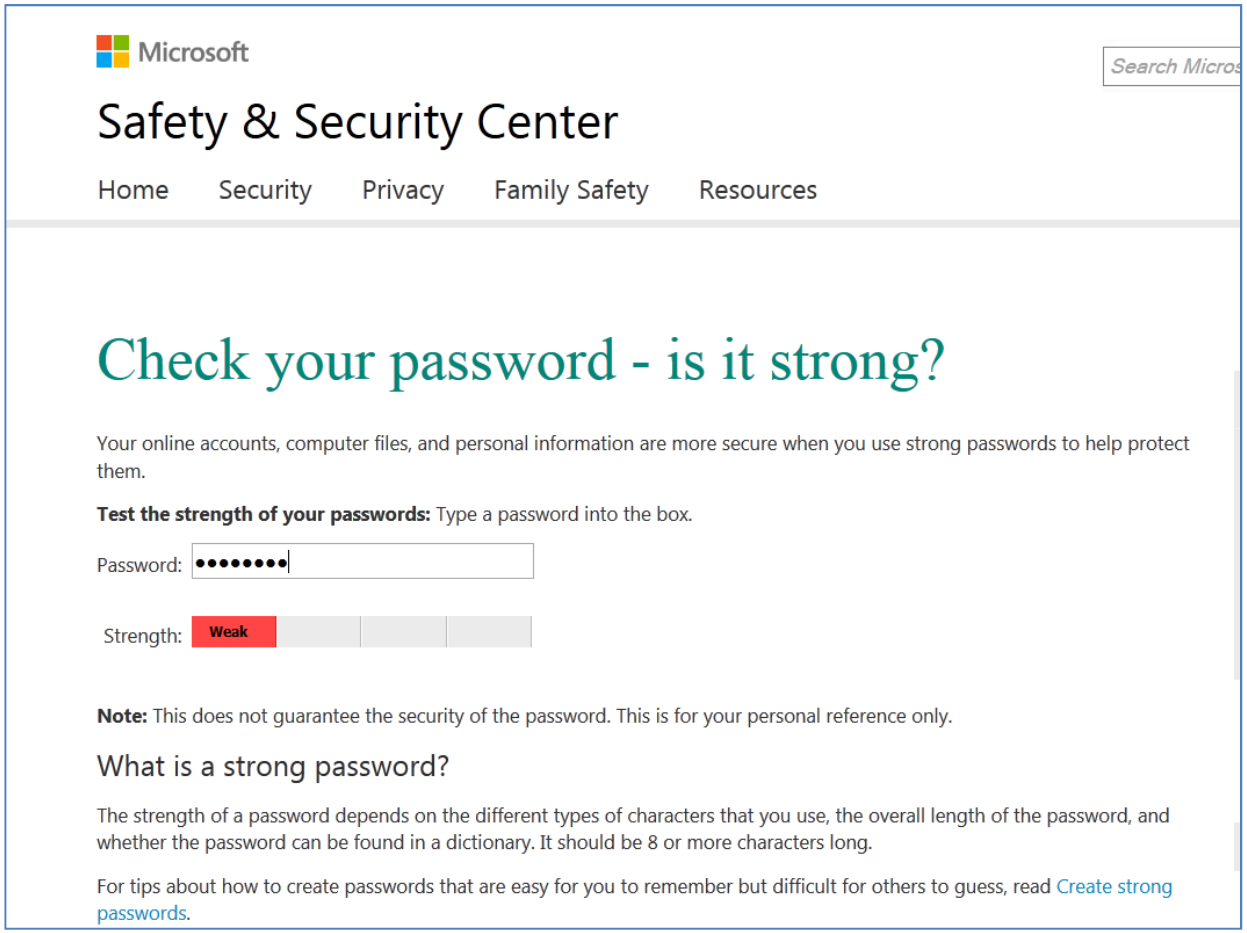
Characteristics of strong passwords

- Strong Passwords
  - contain at least one of each of the following:
    - digit (0..9)
    - letter (a..Z)
    - punctuation symbol (e.g., !)
    - control character
  - are based on a verse (e.g., passphrase) from an obscure work where the password is formed from the characters in the verse

https://xkcd.com/936/

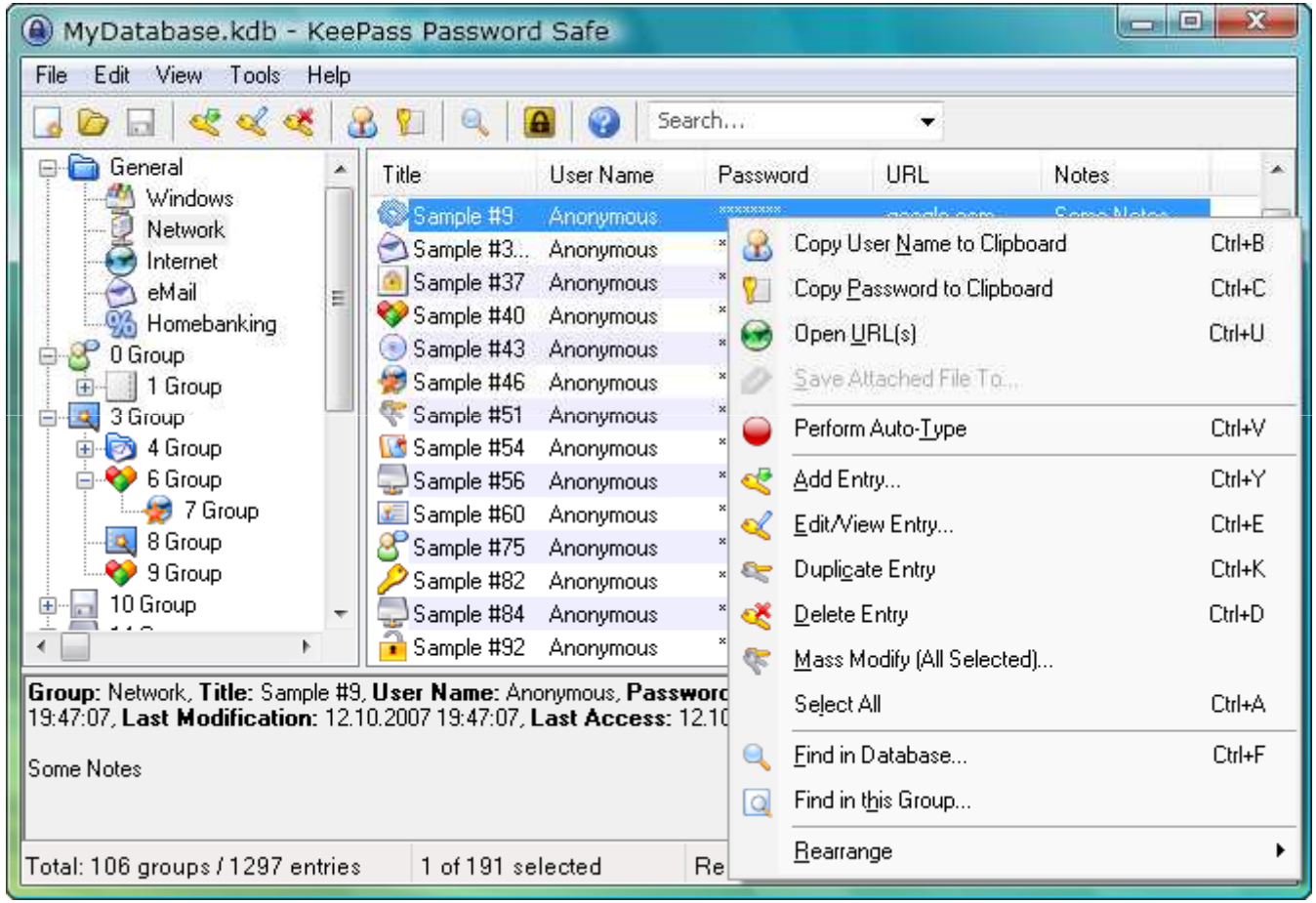


Test your password!



https://www.microsoft.com/security/pc-security/password-checker.aspx

Password Manager

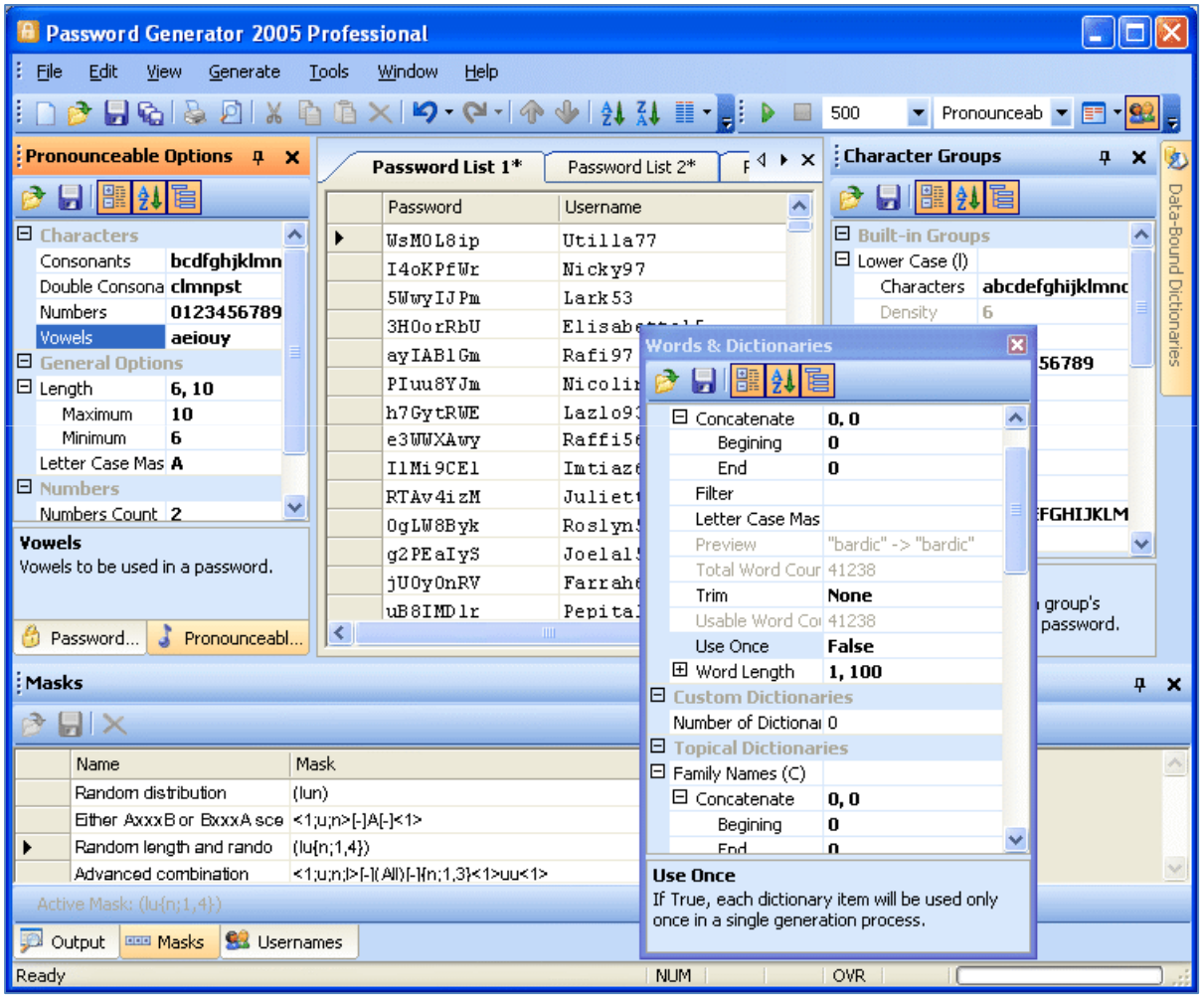


http://keepass.info/

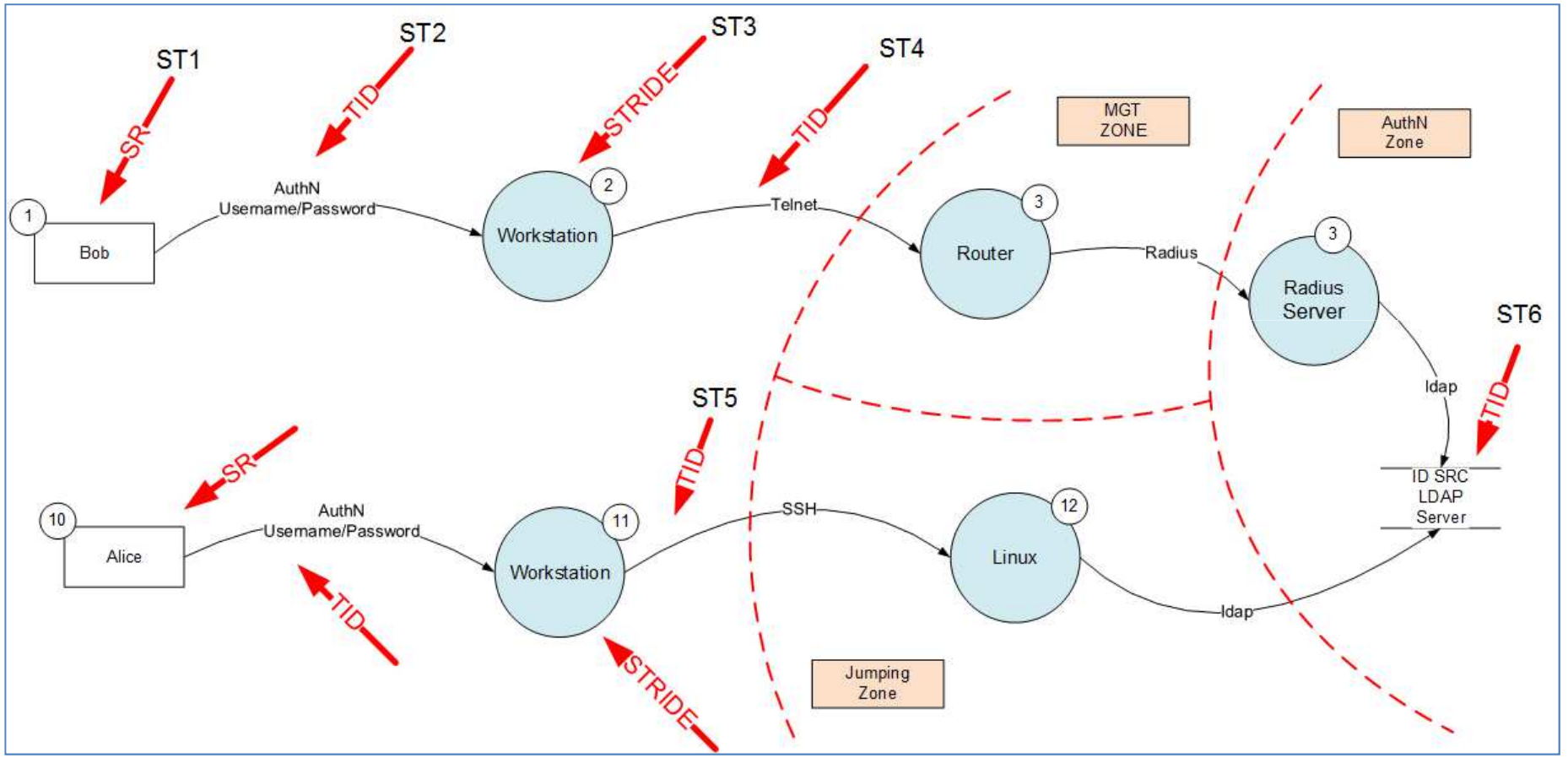
Password Manager

http://passwordsafe.sourceforge.net/

Password Generator



Threat Model AuthN 1FA



Password / Threats

- Man In The Middle Attacks
- Phishing Attacks
- Pharming Attacks
  - DNS Cache Poisoning
- Trojan Attacks
- Man-in-the-Phone Attacks (Man-in-the-Mobile/MitMo Attacks)
- Man-in-the-Browser Attacks
- Browser Poisoning
- Password Sniffing
- Brute Force Attack
- Dictionary Attacks
- Default Password
- Social Engineering

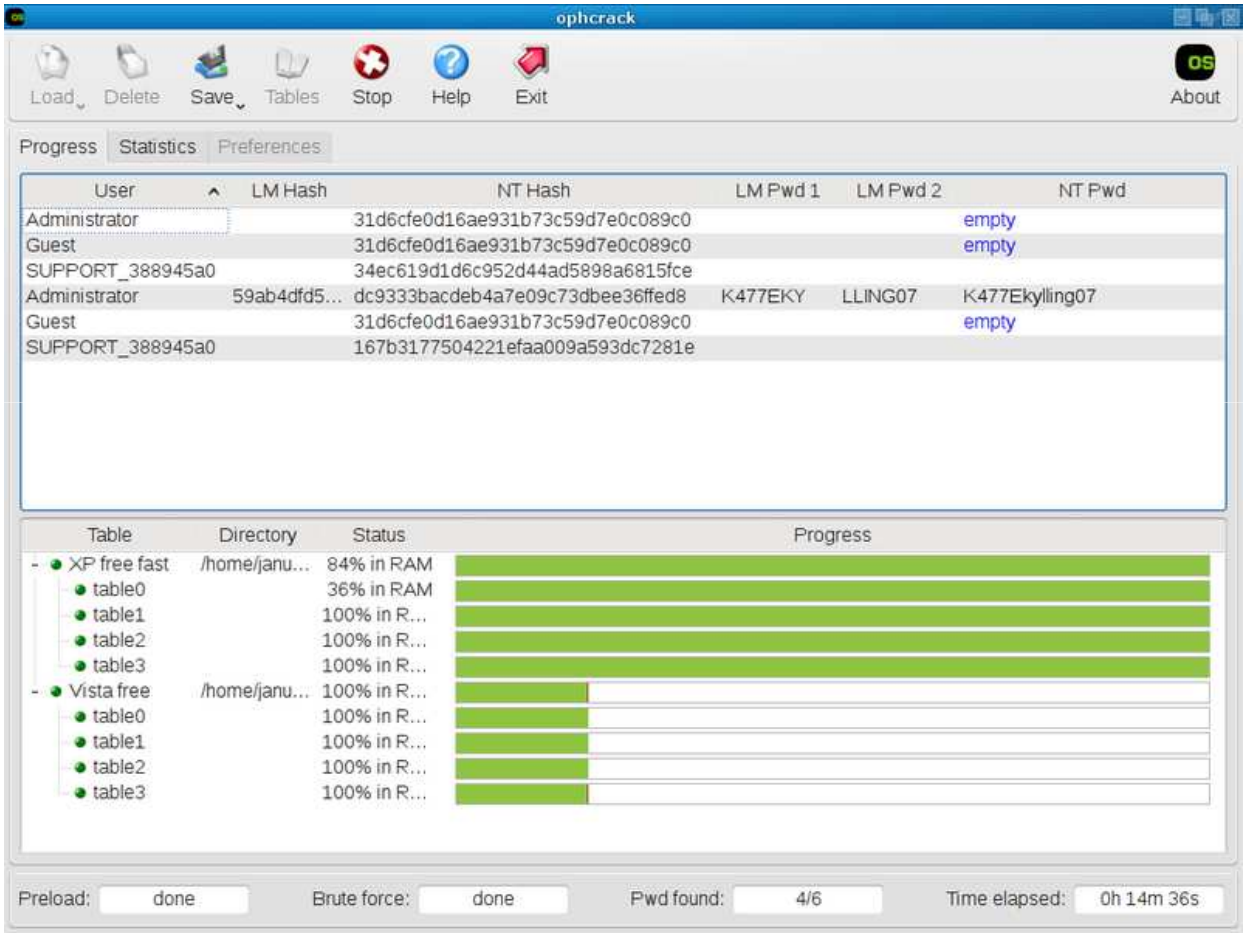
Password Cracking Tools

- Caen & Abel
- John the Ripper
- L0phtCrack
- Ophcrack
- THC hydra
- Aircrack (WEP/WPA cracking tool)
- Etc.

Rainbow table

- A rainbow table is a precomputed table for reversing cryptographic hash functions, usually for cracking password hashes.

Ophcrack



Defense against rainbow tables

- A rainbow table is ineffective against one-way hashes that include salts

```
saltedhash(password) = hash(password+salt)
```

Or

```
saltedhash(password) = hash(hash(password)+salt)
```

Password Storage Cheat Sheet

- Password Storage Rules
  - Rule 1: Use An Adaptive One-Way Function
    - bcrypt, PBKDF2 or scrypt
  - Rule 2: Use a Long Cryptographically Random Per-User Salt
  - Rule 3: Iterate the hash
  - Rule 4 : Encrypt the Hash Data With a Keyed Algorithm

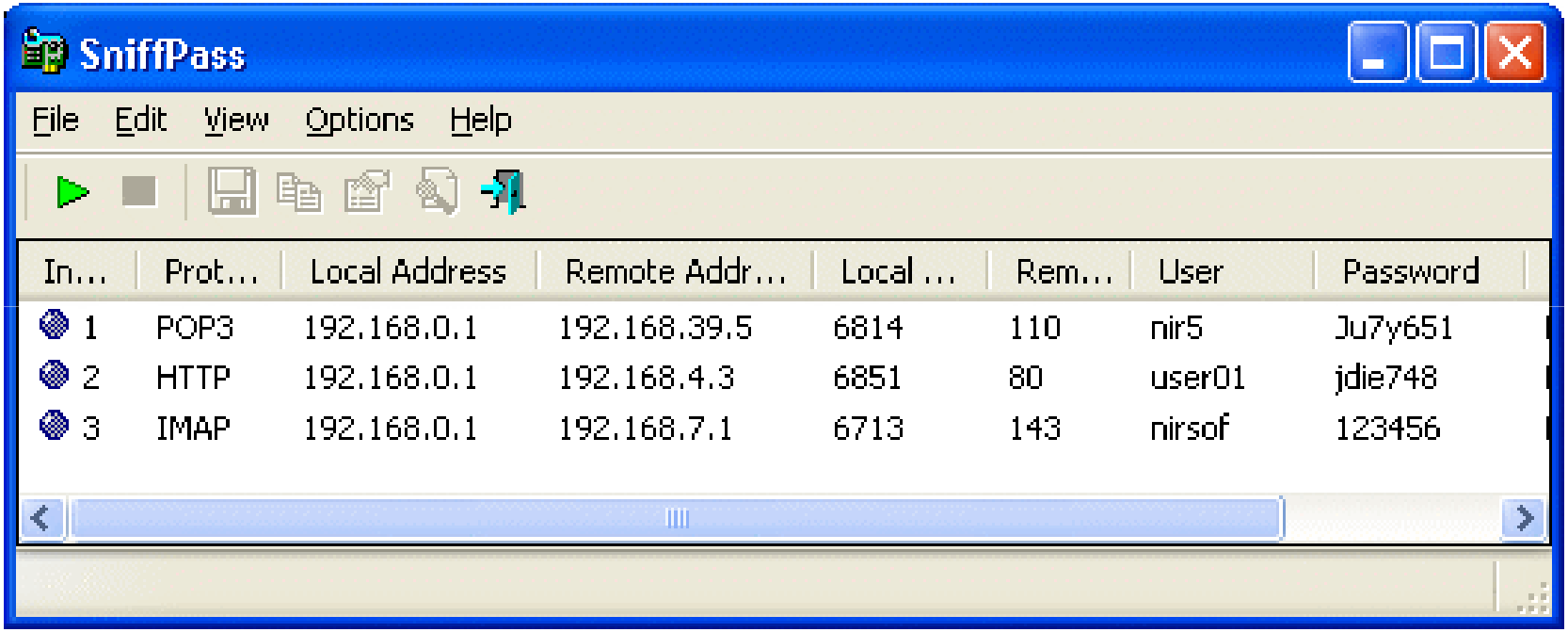
https://www.owasp.org/index.php/Password_Storage_Cheat_Sheet

Hashcat / GPU

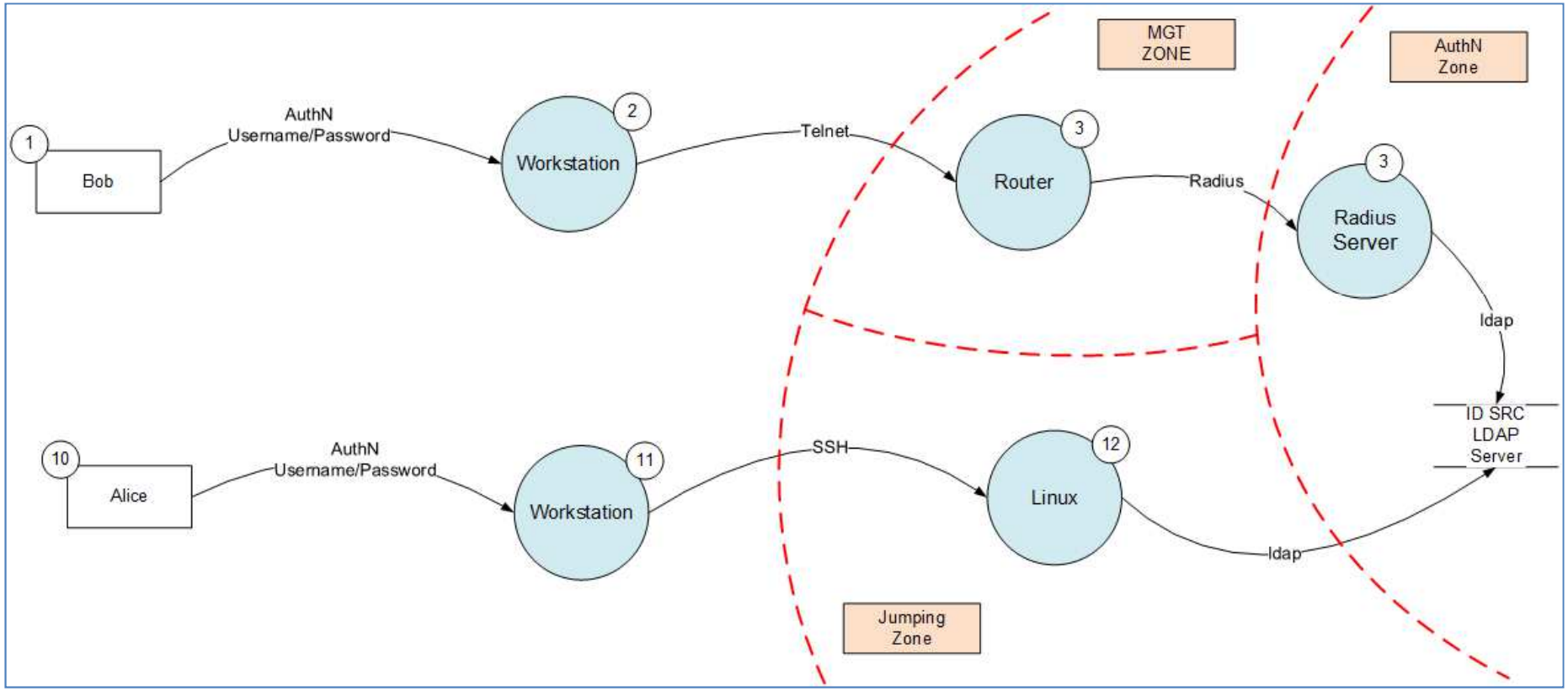
- 25-GPU cluster cracks every standard Windows password in <6 hours
  - It achieves the 350 billion-guess-per-second speed when cracking password hashes generated by the NTLM cryptographic algorithm that Microsoft has included in every version of Windows since Server 2003.

http://arstechnica.com/security/2012/12/25-gpu-cluster-cracks-every-standard-windows-password-in-6-hours/

Password sniffing



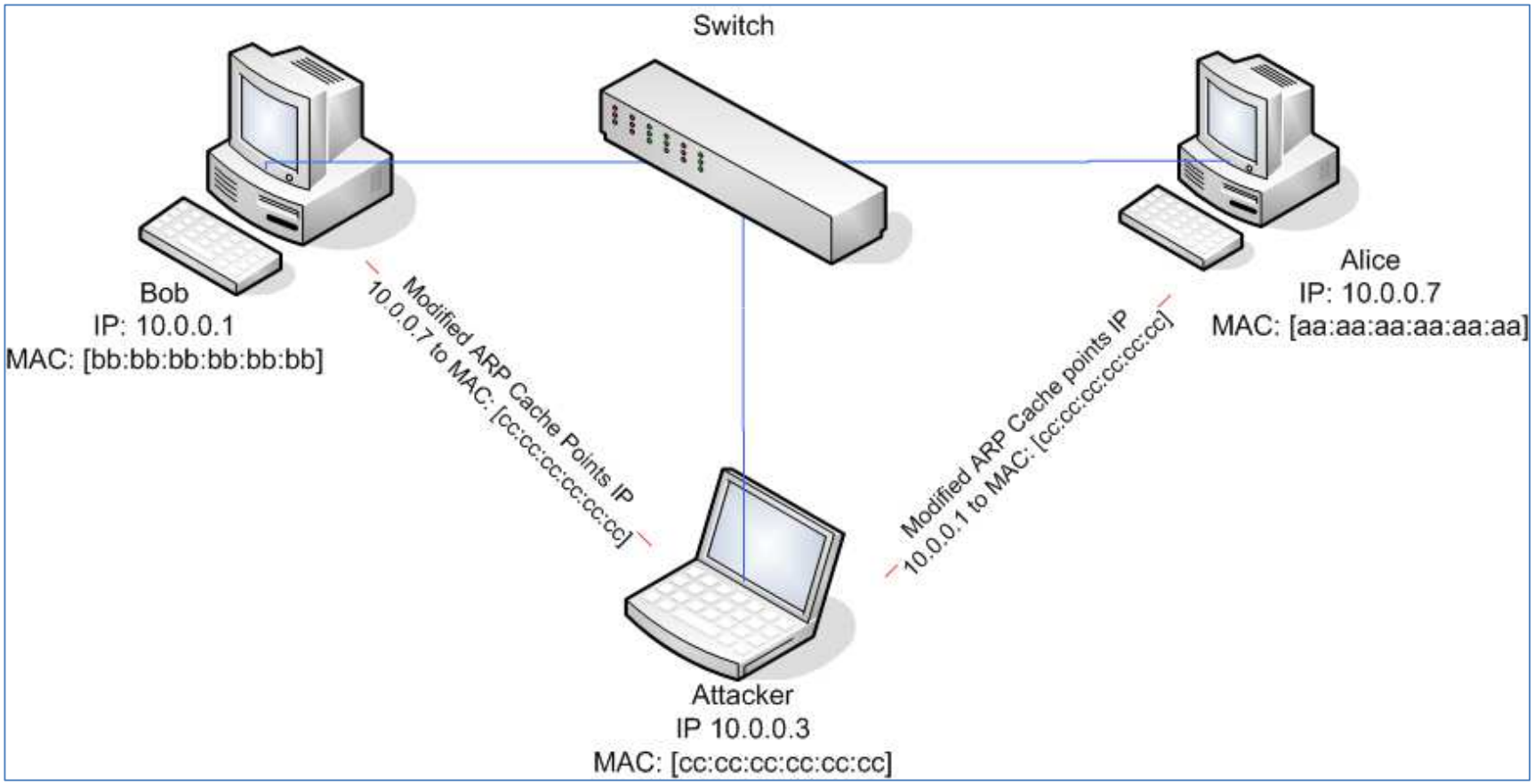
DFD – Weak Protocol (Telnet)



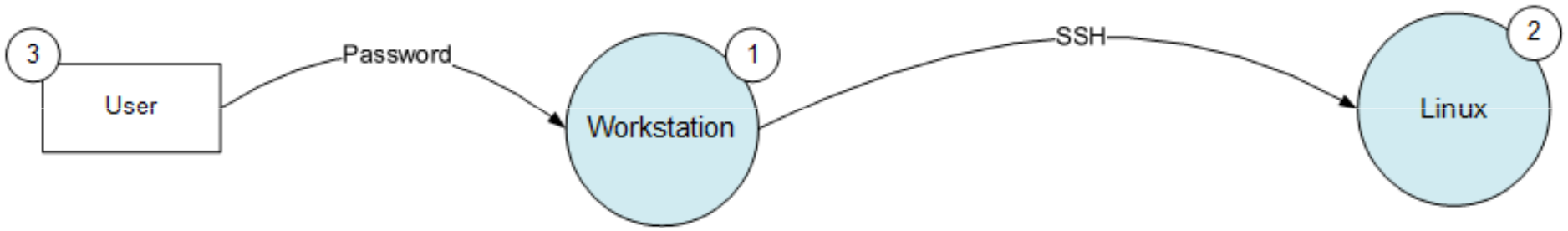
Weak protocols

- Telnet
- FTP
- IMAP
- POP3
- LDAP
- Etc.

ARP Spoofing

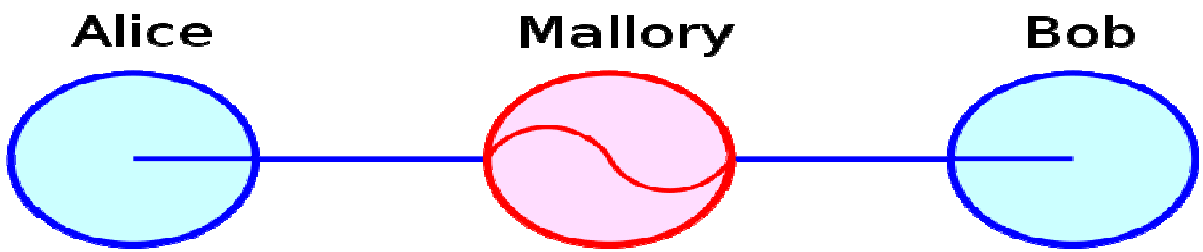


DFD - SSH



Man-in-the-middle attack

- often abbreviated
  - MITM, MitM, MIM, MiM, MITMA



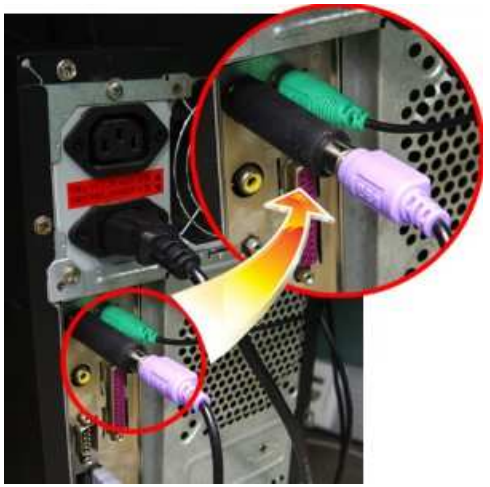
Man-in-the-middle attack

- Ettercap
- SSLStrip
- SSLSniff
- Mallory
- Etc.

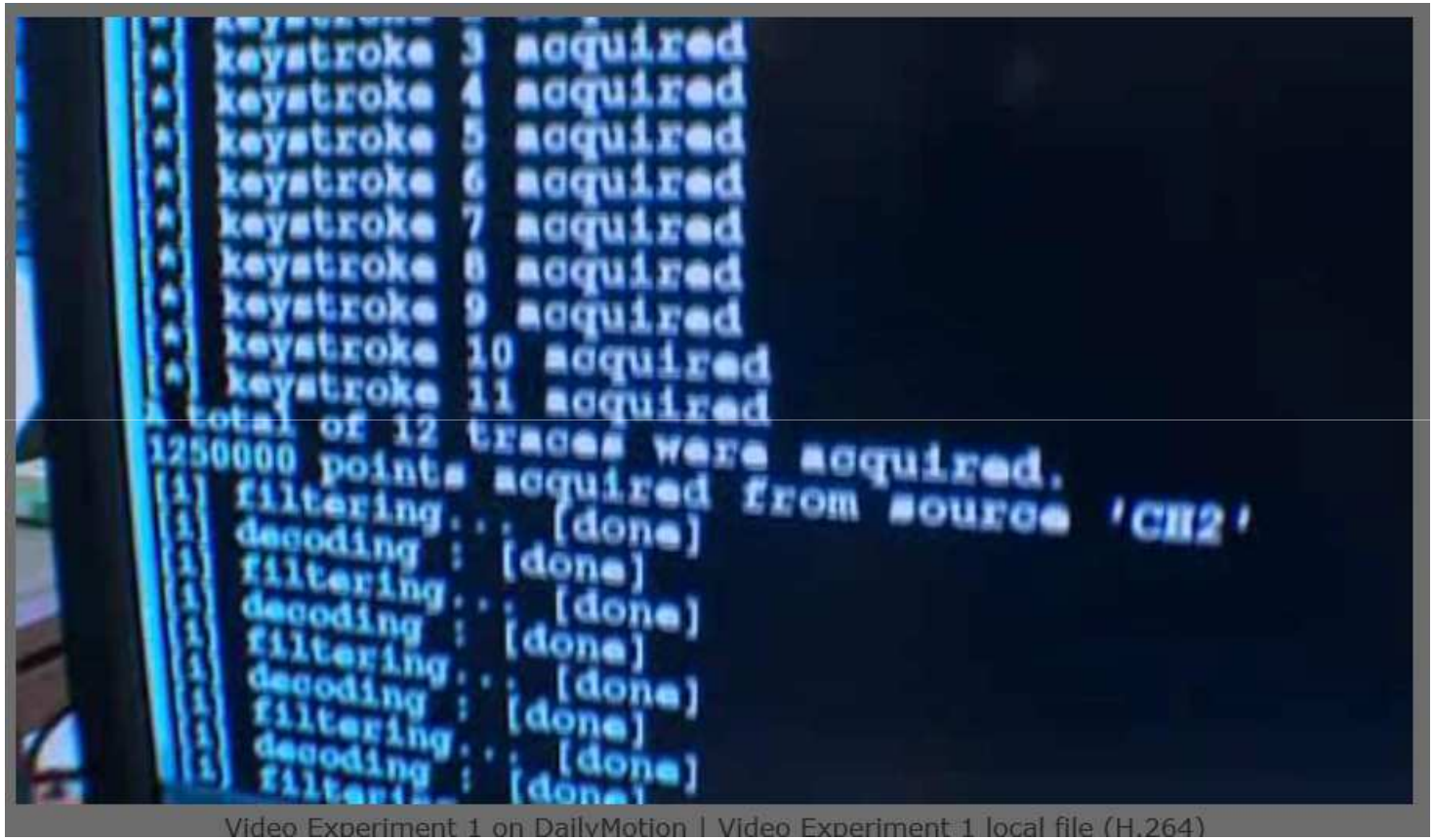
Keylogger / Keystroke logging

- Software-based keyloggers
  - Malware
  - Mobile

- Hardware-based keyloggers



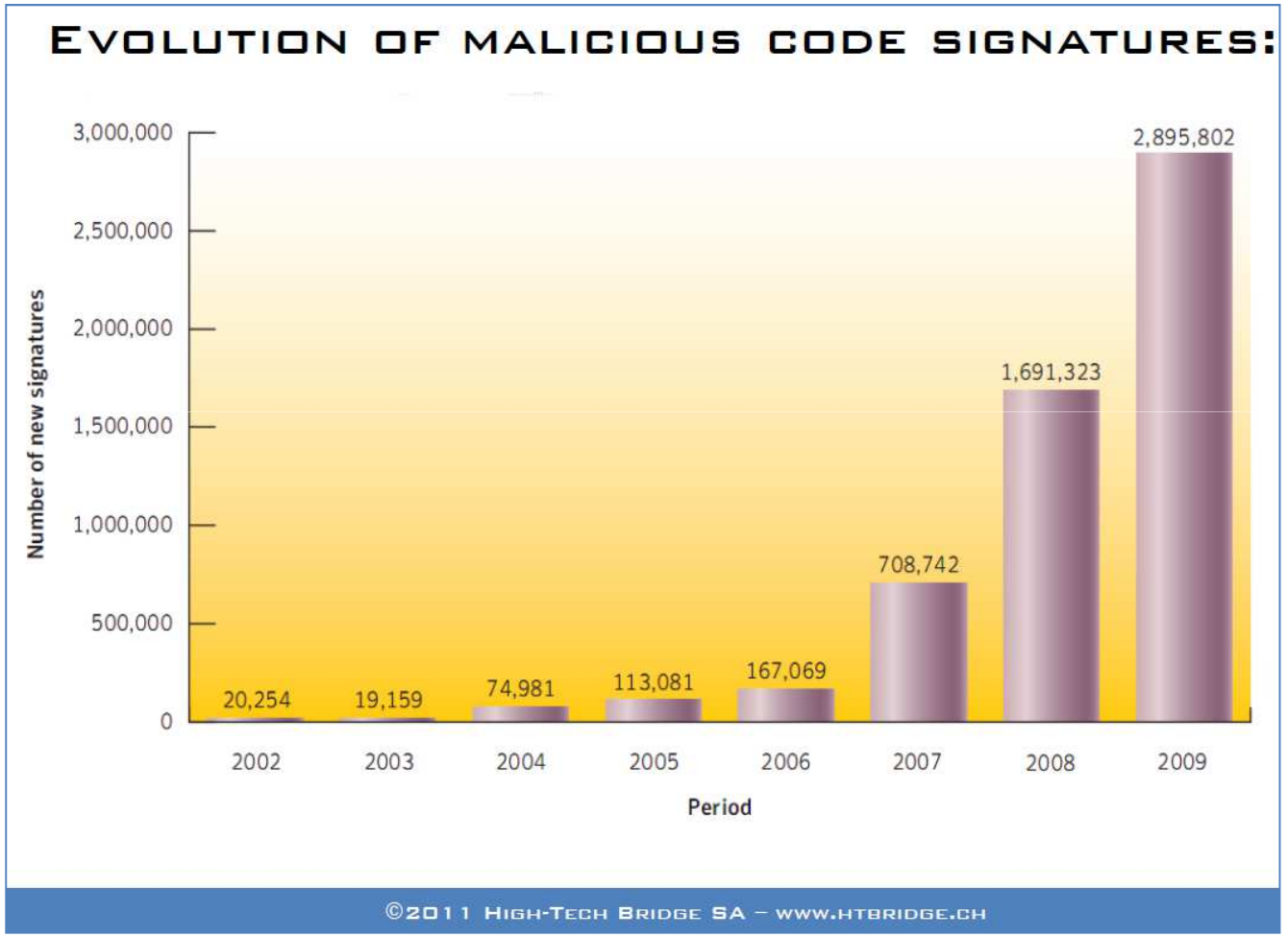
Wireless sniffing – TEMPEST



Video Experiment 1 on DailyMotion | Video Experiment 1 local file (H.264)

http://lasecwww.epfl.ch/keyboard/

Malicious Code Evolution



Malware

Zeus

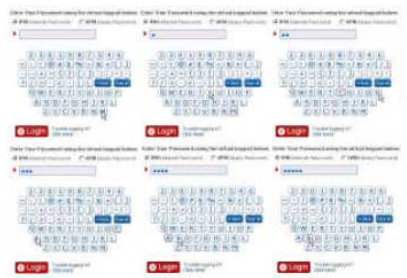


Figure 22: Virtual keyboard session, retrieved images

The anatomy of the Gameover Zeus variant

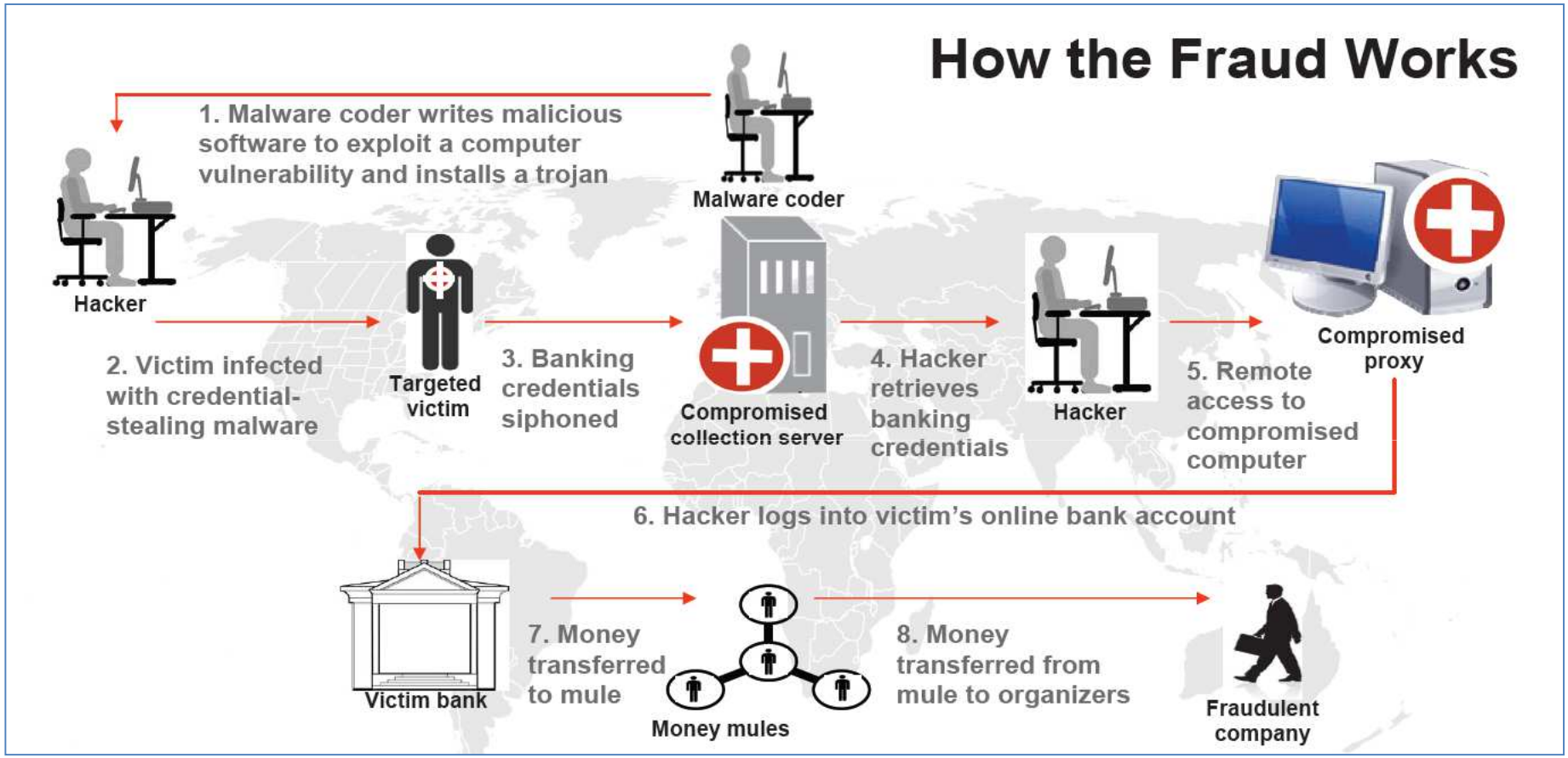
Posted on 11.01.2012

BOOKMARK

The "Gameover" malware is a relatively new, "private" version of ZeuS. Support for the distributed command and control (C2) tools, integrated into the ZeuS botnet, were implemented at the request of one of the "private" clients of the ZeuS author.

Distributed C2 is a feature which was originally considered by the malware author in the ZeuS 1.4/2.0 beta program, but it was dropped from the final 2.0.x release because lack of demand among ZeuS customers in the face of significant coding and testing time. It was put back in as a feature during the recent, ongoing 2.2/3.0 beta program.

The "Gameover" version of Zeus also supports the use of complex web injections that allow the attacker to perform Man-in-the-Browser (MITB) attacks to bypass multi-factor authentication mechanisms. The ZeuS author has also rolled a Distributed Denial of Service (DDoS) component into the Gameover bundle.


How the Fraud Works
1. Malware coder writes malicious software to exploit a computer vulnerability and installs a trojan
Malware coder
Hacker
2. Victim infected with credential-stealing malware
Targeted victim
3. Banking credentials siphoned
Compromised collection server
4. Hacker retrieves banking credentials
Hacker
5. Remote access to compromised computer
Compromised proxy
6. Hacker logs into victim's online bank account
Victim bank
7. Money transferred to mule
Money mules
8. Money transferred from mule to organizers
Fraudulent company

Default Password

1. Alteon - *ACEswitch*

Version	180e
Method	HTTP
User ID	admin
Password	admin
Level	Administrator

2. Alteon - *ACEswitch*

Version	180e
Method	Telnet
User ID	admin
Password	(none)

One Time Password - OTP

Strong AuthN OTP

OTP Technology / Standards

- Based on a shared secret Key (symmetric Crypto)
- Approach
  - Time Based OTP
  - Event Based OTP
  - Challenge Response OTP
  - Out-of-band OTP
  - Transaction Signing OTP
  - Others
- Standards
  - OATH

Time Based OTP

K=Secret Key / Seed

HMAC

OTP

T=UTC Time

Event Based OTP

K=Secret Key / Seed

HMAC

OTP

C = Counter

OTP Challenge Response Based

K=Secret Key / Seed

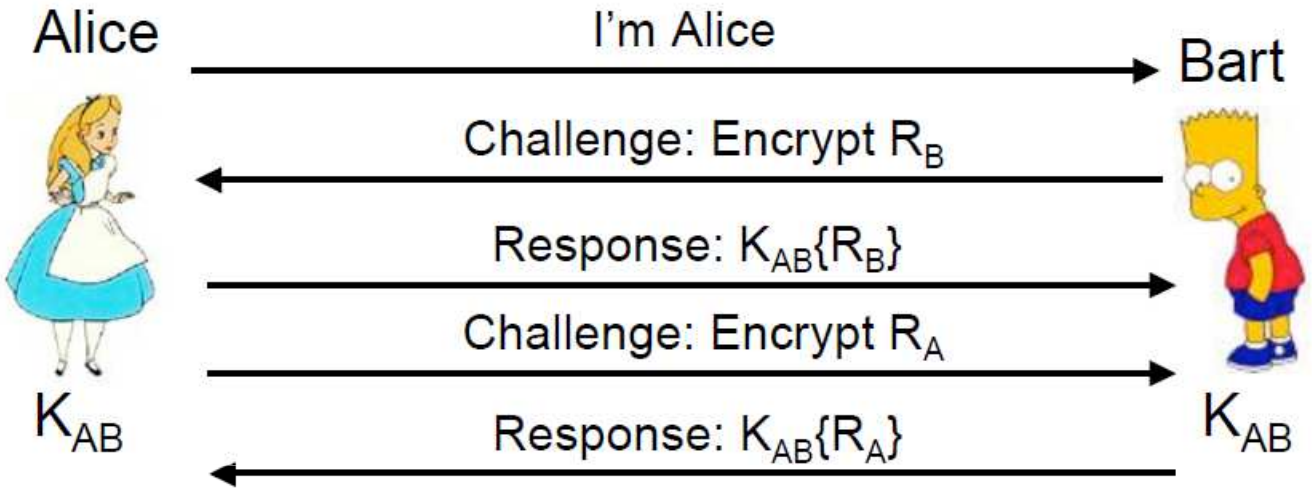
HASH Function

OTP Challenge

nonce

```
CLIENT                                          SERVER
(PROVER)                                        VERIFIER)
   |                                               |
   |   Verifier sends challenge to prover          |
   |   Challenge = Q                               |
   |<----------------------------------------------|
   |                                               |
   |   Prover Computes Response                    |
   |   R = OCRA(K, {[C] | Q | [P | S | T]})       |
   |   Prover sends Response = R                   |
   |---------------------------------------------->|
   |                                               |
   |  Verifier Validates Response                  |
   |  If Response is valid, Server sends OK        |
   |  If Response is not,  Server sends NOK        |
   |<----------------------------------------------|
   |                                               |
```

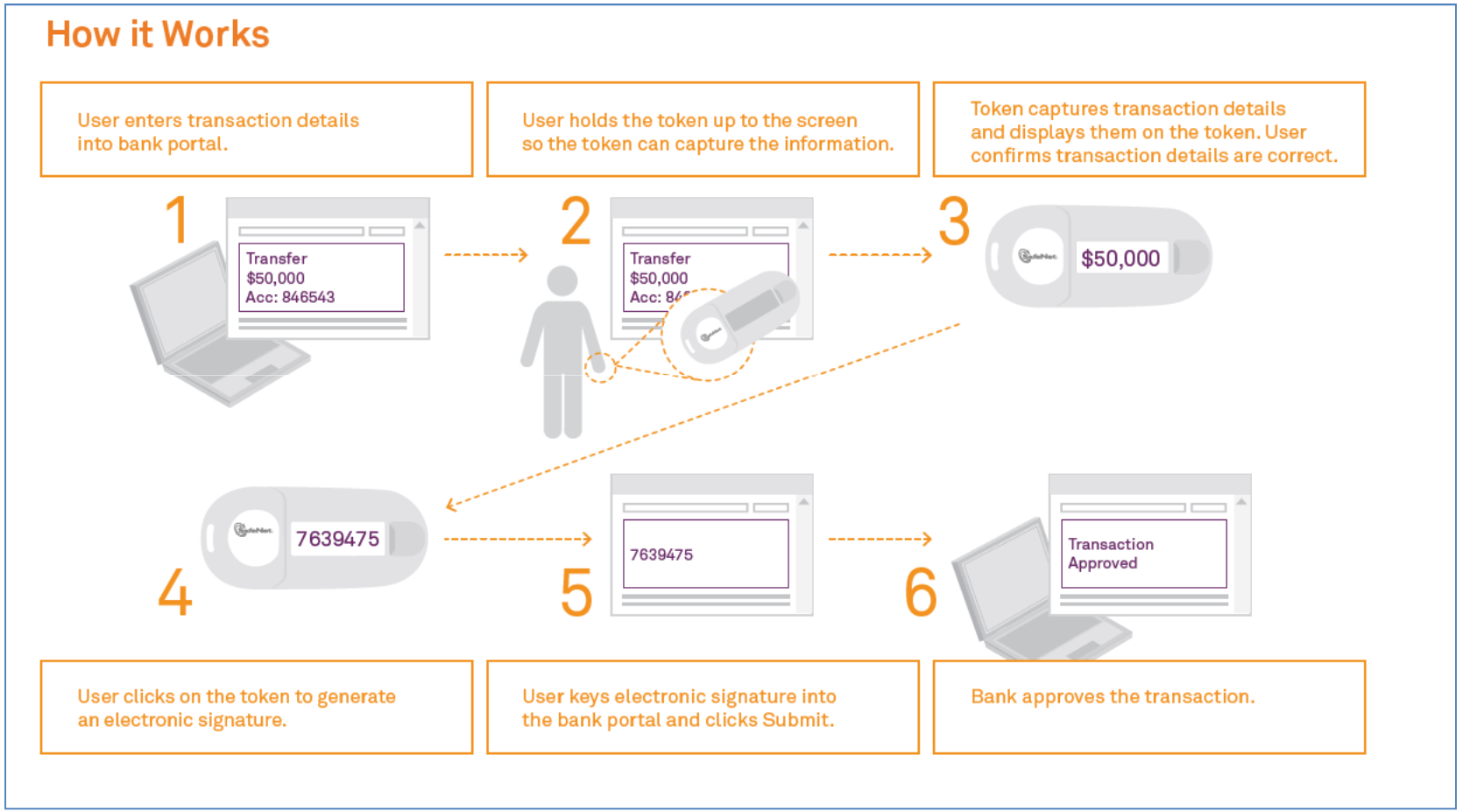
Recap: Challenge Response



- Protocol doesn't reveal the secret.
- *Challenge/Response*
  - Bart requests proof that Alice knows the secret
  - Alice requires proof from Bart
  - R_A and R_B are randomly generated numbers

Source= CSE331: Introduction to Networks and Security

Transaction Signing OTP



Source= Safenet

Token OTP pin protected

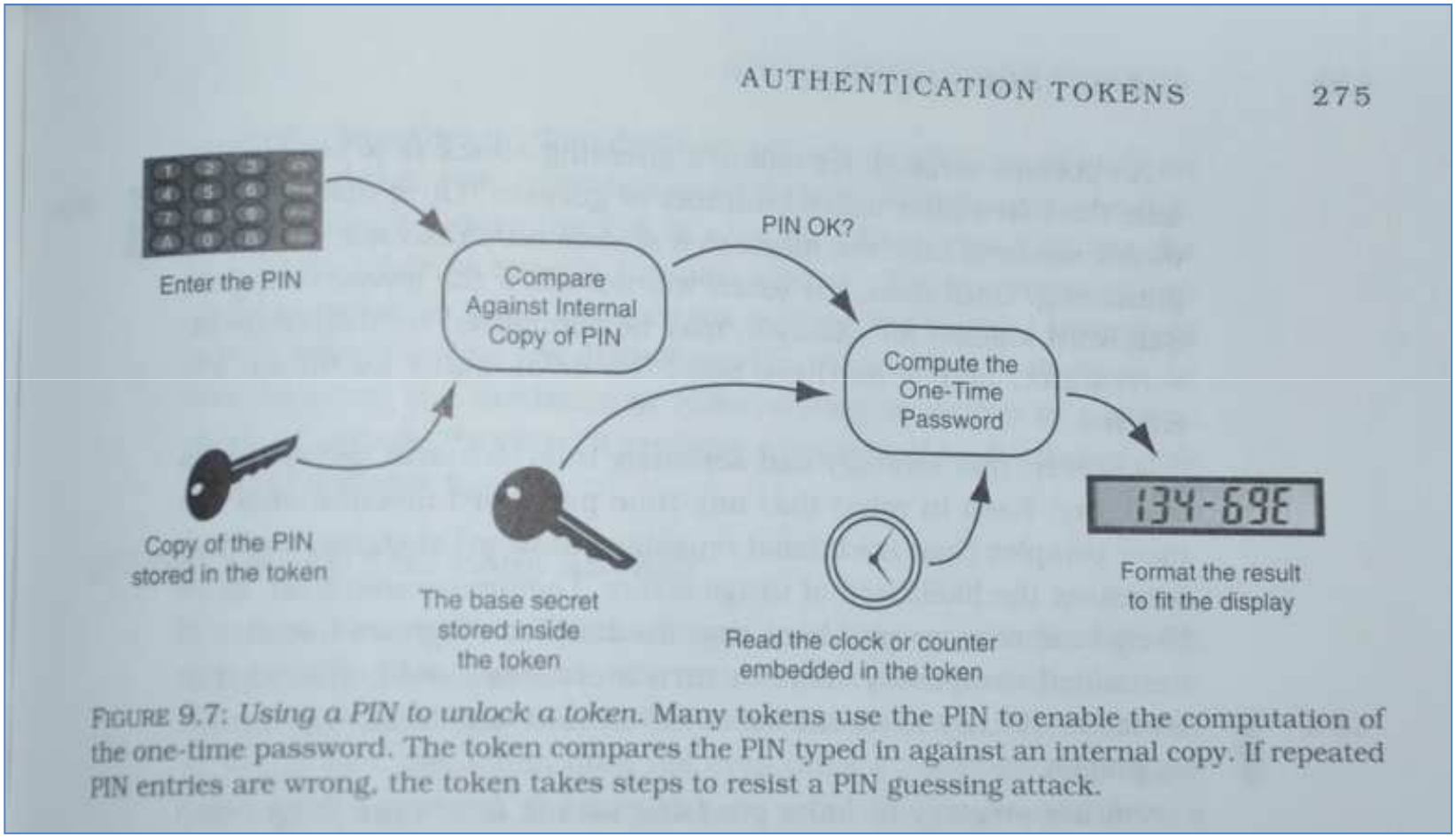


FIGURE 9.7: *Using a PIN to unlock a token.* Many tokens use the PIN to enable the computation of the one-time password. The token compares the PIN typed in against an internal copy. If repeated PIN entries are wrong, the token takes steps to resist a PIN guessing attack.

Source: Richard E. Smith / Authentication

Token OTP pin protected

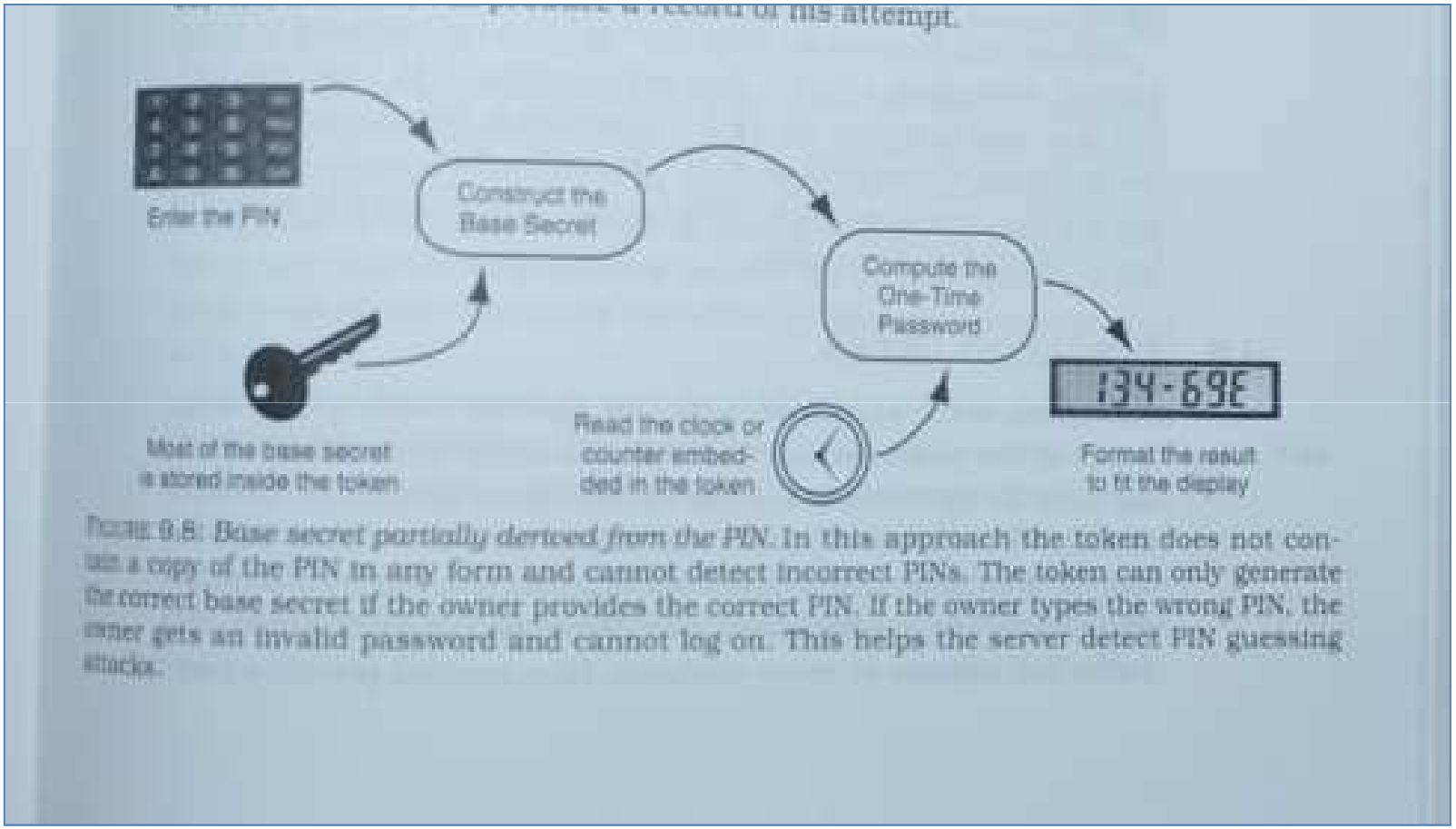


FIGURE 9.8: *Base secret partially derived from the PIN.* In this approach the token does not contain a copy of the PIN in any form and cannot detect incorrect PINs. The token can only generate the correct base secret if the owner provides the correct PIN. If the owner types the wrong PIN, the owner gets an invalid password and cannot log on. This helps the server detect PIN guessing attacks.

Source: Richard E. Smith / Authentication

Others OTP

- SMS OTP
- TAN
- paper-based OTP
- Bingo Card
- Etc.

Out-of-band - SMS OTP

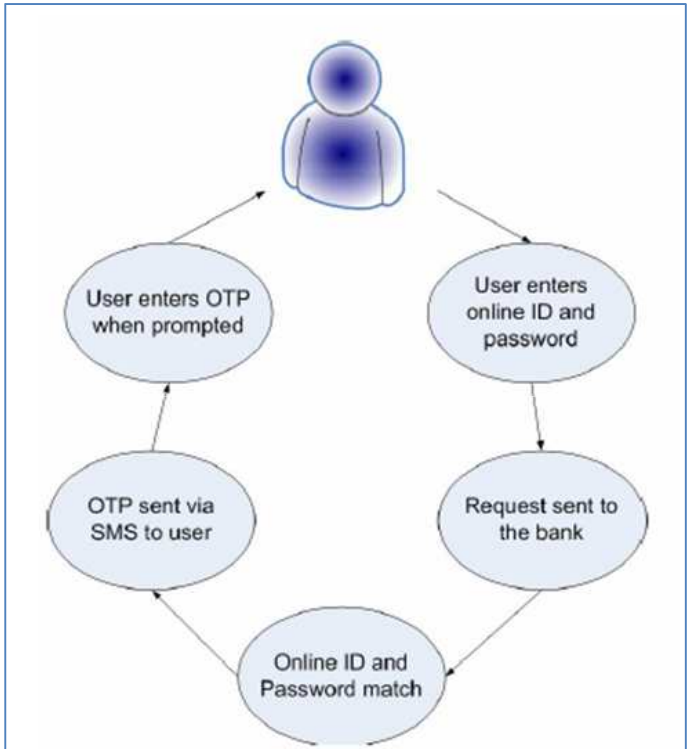


Figure 1: Using an SMS based One-Time Password

Out-of-band - TAN OTP

001	~~2455~~	021	2455
002	~~4389~~	022	4389
003	~~8953~~	023	8953
004	~~0583~~	024	0583
005	~~3281~~	025	3281
006	~~1049~~	026	1049
007	~~7281~~	027	7281
008	~~2988~~	028	2988
009	~~9723~~	029	9723
010	~~2589~~	030	2589

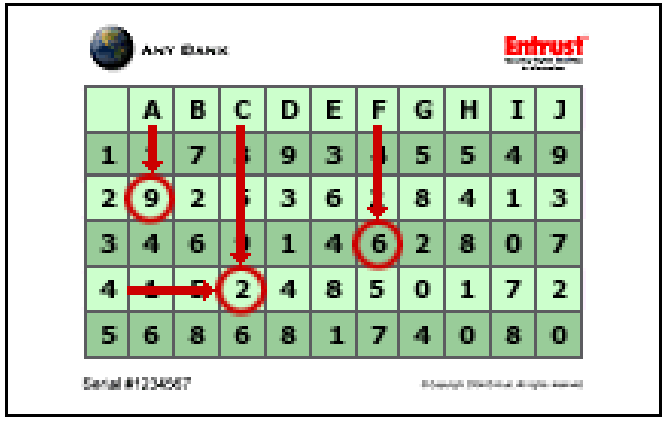
paper-based OTP

Paper token for HOTP token S/N : 3132333435
How to use? Read from left to right and when you use a token, strikethrough the used value.
Generated by paper token - http://www.foo.be/paper-token/ at Sun Jun 6 19:11:01 2010

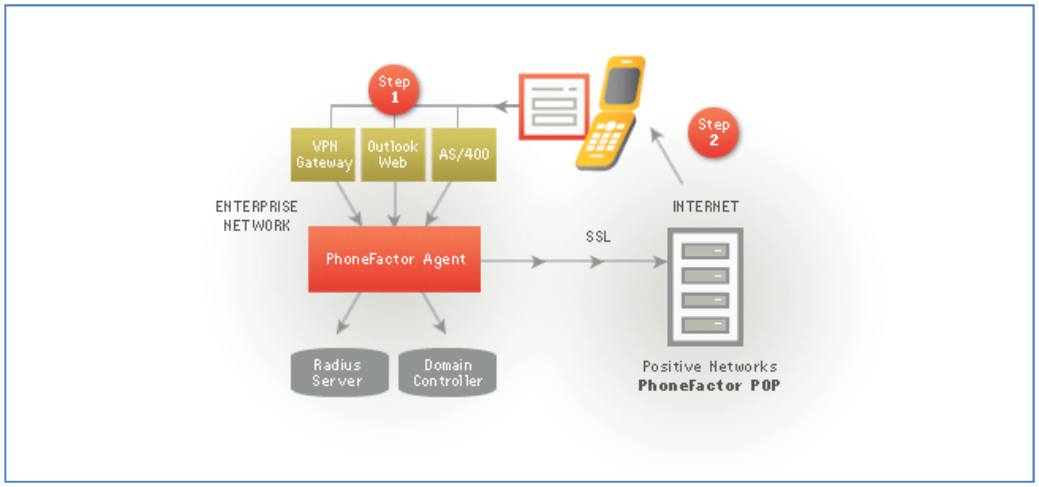
755224 287082 359152 969429 338314 254676 287922 162583 399871 520489 403154 481090 868912 736127
229903 436521 186581 447589 903435 578337 328281 191635 184416 574561 797908 396619 122382 939082
908316 316591 026920 523596 370250 841346 749439 037211 003784 520231 521952 619416 268376 471723
435478 303194 000152 287422 318298 098238 039329 710717 528155 980838 249088 354406 399156 478026
294892 941415 964363 083773 864257 719632 005080 925505 317632 088627 024418 954526 036679 864060
569881 029459 486963 559613 202372 705686 272974 379493 839877 393669 863623 198167 935444 108405
305499 563707 447439 332099 087812 032830 811649 190372 458549 202468 722946 047817 229689 430056
289357 516516 295165 329376 629694 378717 694769 804168 290960 207438 466040 012238 863891 133688
702014 438906 780546 240957 862652 926140 455436 587786 929786 849648 577879 033991 390600 670144
986293 307470 425216 334228 156835 651889 570405 186928 591313 807552 347330 844442 656754 759308
679732 636068 117604 334243 273557 415001 415507 719508 888238 449000 072172 072953 801020 594526
393059 678706 141555 215923 711384 603868 953080 656434 951858 133817 047086 323790 747772 738595
370733 141671 844986 152983 048765 663651 176699 944323 528241 023376 934556 519205 897140 790138
406554 536163 650833 473091 879076 443996 712288 915604 666078 427801 471610 682261 819211 559282
340234 520705 117750 492354 466290 462985 107630 113587 714185 869934 639667 027999 892527 750171

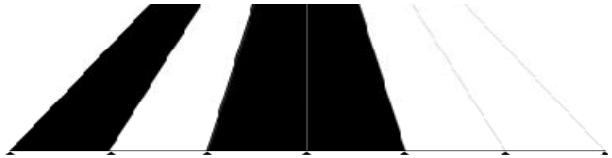
https://github.com/adulau/paper-token

Bingo Card OTP



Other[s] OTP technologies…





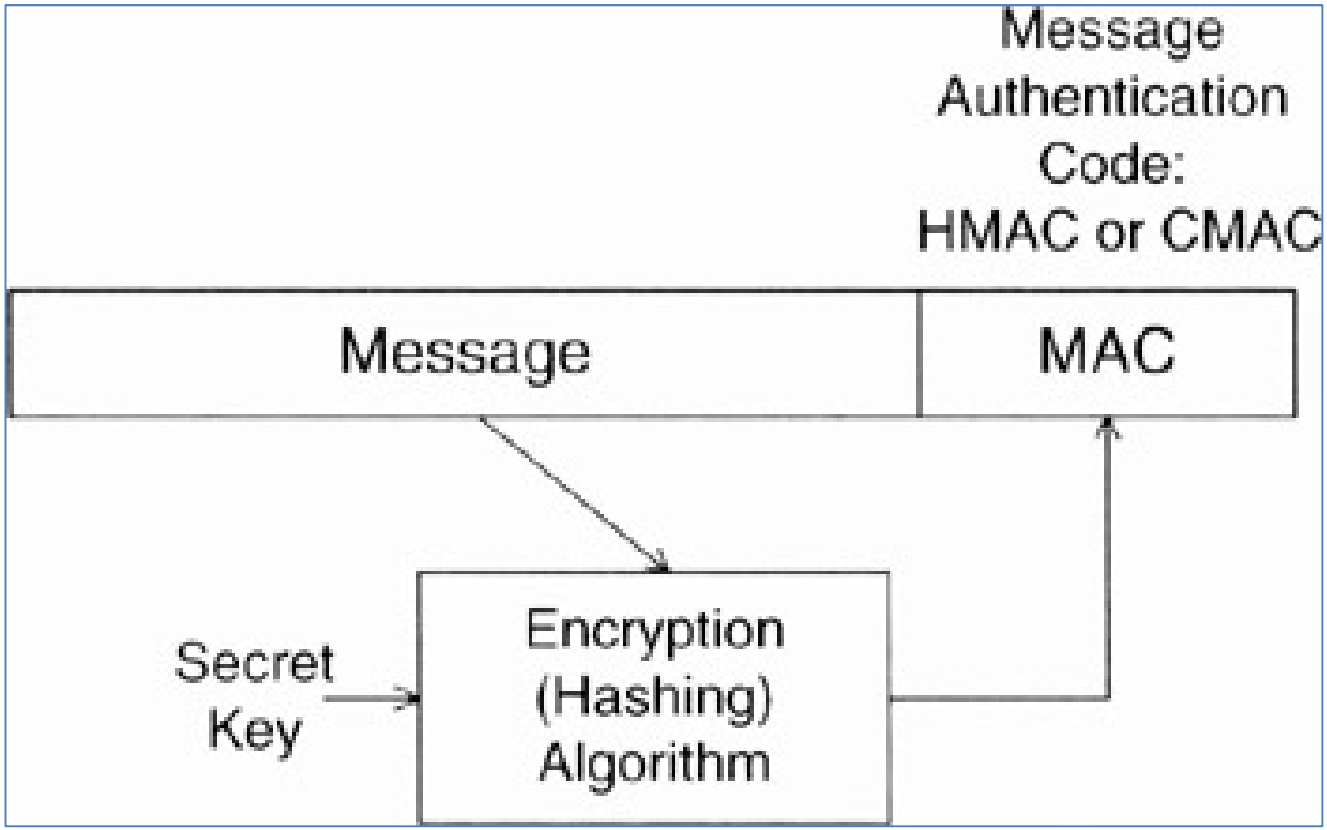
“Flicker code” Generator Software that converts already encrypted data into optical screen animation

initiative for open authentication

OTP / OATH standards

Authentication Methods

HMAC – 101 (Keyed-Hashing for Message Authentication)



http://www.ietf.org/rfc/rfc2104.txt

OATH - Authentication Methods

- HOTP: An HMAC-Based OTP Algorithm (RFC 4226)
- TOTP - Time-based One-time Password Algorithm (RFC 6238)
- OCRA - OATH Challenge/Response Algorithms Specification (RFC 6287)

HOTP: An HMAC-Based One-Time Password Algorithm

- RFC 4226
- http://www.ietf.org/rfc/rfc4226.txt
- Event Based OTP
- Use HMAC: Keyed-Hashing for Message Authentication (RFC 2104)

HOTP – Crypto 101

Symbol	Represents
C	8-byte counter value, the moving factor. This counter MUST be synchronized between the HOTP generator (client) and the HOTP validator (server).
K	shared secret between client and server; each HOTP generator has a different and unique secret K.
T	throttling parameter: the server will refuse connections from a user after T unsuccessful authentication attempts.

HOTP – Crypto 101

The HOTP algorithm is based on an increasing counter value and a static symmetric key known only to the token and the validation service. In order to create the HOTP value, we will use the HMAC-SHA-1 algorithm, as defined in RFC 2104 [BCK2].

As the output of the HMAC-SHA-1 calculation is 160 bits, we must truncate this value to something that can be easily entered by a user.

```
HOTP(K,C) = Truncate(HMAC-SHA-1(K,C))
```

Where:

- Truncate represents the function that converts an HMAC-SHA-1 value into an HOTP value as defined in Section 5.3.

TOTP - Time-based One-time Password Algorithm

- RFC 6238
- http://www.ietf.org/rfc/rfc6238.txt
- Time Based OTP
- Use HMAC: Keyed-Hashing for Message Authentication (RFC 2104)

TOTP – Crypto 101

4. TOTP Algorithm

This variant of the HOTP algorithm specifies the calculation of a one-time password value, based on a representation of the counter as a time factor.

4.1. Notations

- X represents the time step in seconds (default value X = 30 seconds) and is a system parameter.
- T0 is the Unix time to start counting time steps (default value is 0, i.e., the Unix epoch) and is also a system parameter.

4.2. Description

Basically, we define TOTP as TOTP = HOTP(K, T), where T is an integer and represents the number of time steps between the initial counter time T0 and the current Unix time.

More specifically, T = (Current Unix time - T0) / X, where the default floor function is used in the computation.

For example, with T0 = 0 and Time Step X = 30, T = 1 if the current Unix time is 59 seconds, and T = 2 if the current Unix time is 60 seconds.

Challenge Response OTP

- RFC 6287
- http://www.ietf.org/rfc/rfc6287.txt
- OCRA
- OATH Challenge-Response Algorithm

OCRA – Crypto 101

5. Definition of OCRA

The OATH Challenge-Response Algorithm (OCRA) is a generalization of HOTP with variable data inputs not solely based on an incremented counter and secret key values.

The definition of OCRA requires a cryptographic function, a key K and a set of DataInput parameters. This section first formally introduces OCRA and then introduces the definitions and default values recommended for all parameters.

In a nutshell,

OCRA = CryptoFunction(K, DataInput)

OCRA – Crypto 101

where:

- K: a shared secret key known to both parties
- DataInput: a structure that contains the concatenation of the various input data values defined in details in section 5.1
- CryptoFunction: this is the function performing the OCRA computation from the secret key K and the DataInput material;

CryptoFunction is described in details in Section 5.2

OCRA – Crypto 101

5.2. CryptoFunction

The default CryptoFunction is HOTP-SHA1-6, i.e., the default mode of computation for OCRA is HOTP with the default 6-digit dynamic truncation and a combination of DataInput values as the message to compute the HMAC-SHA1 digest.

We denote t as the length in decimal digits of the truncation output. For instance, if t = 6, then the output of the truncation is a 6-digit (decimal) value.

We define the HOTP family of functions as an extension to HOTP:

1. HOTP-H-t: these are the different possible truncated versions of HOTP, using the dynamic truncation method for extracting an HOTP value from the HMAC output

2. We will denote HOTP-H-t as the realization of an HOTP function that uses an HMAC function with the hash function H, and the dynamic truncation as described in [RFC4226] to extract a t-digit value

3. t=0 means that no truncation is performed and the full HMAC value is used for authentication purposes

OATH module 1/2

- http://packages.debian.org/source/testing/oath-toolkit
- https://pypi.python.org/pypi/oath/1.0
- http://www.nongnu.org/oath-toolkit/
- https://github.com/jennings/OATH.Net
- http://search.cpan.org/~sifukurt/Authen-OATH-v1.0.0/lib/Authen/OATH.pm
- http://code.google.com/p/mod-authn-otp/
- https://code.google.com/p/oathtoken/
- http://code.google.com/p/oathtoken/wiki/WebProvisioning

OATH module 2/2

- http://freecode.com/projects/linotp
- http://sourceforge.net/projects/rcdevs-openotp/
- http://www.multiotp.net/
- http://www.rcdevs.com/products/openotp/
- http://blog.josefsson.org/2011/01/20/introducing-the-oath-toolkit/
- http://www.linotp.org/

MobileOTP

- Based on MD5
- Time Based OTP
- http://motp.sourceforge.net/
- http://security.edu.pl/motp-as/login.php

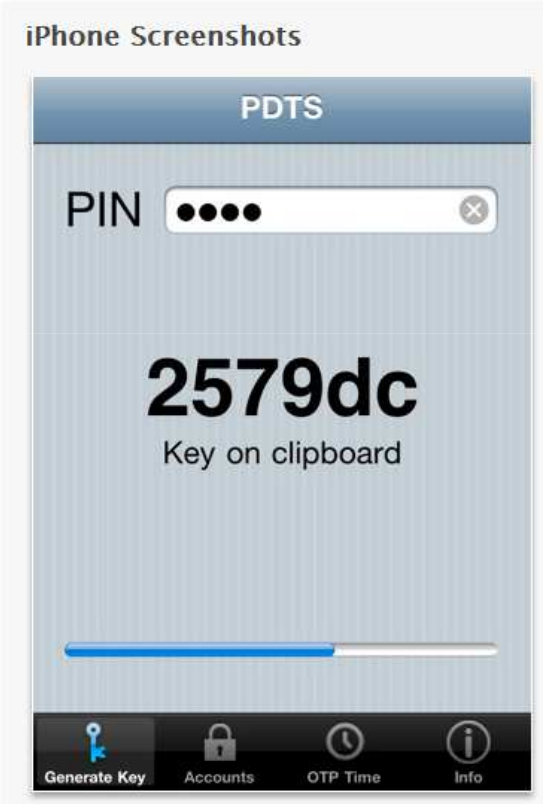
OTP solution

OTP AuthN


swiss sign
AUTHORISED SIGNATURE

OTP Token: Software vs Hardware ?

Software OTP for Smartphone



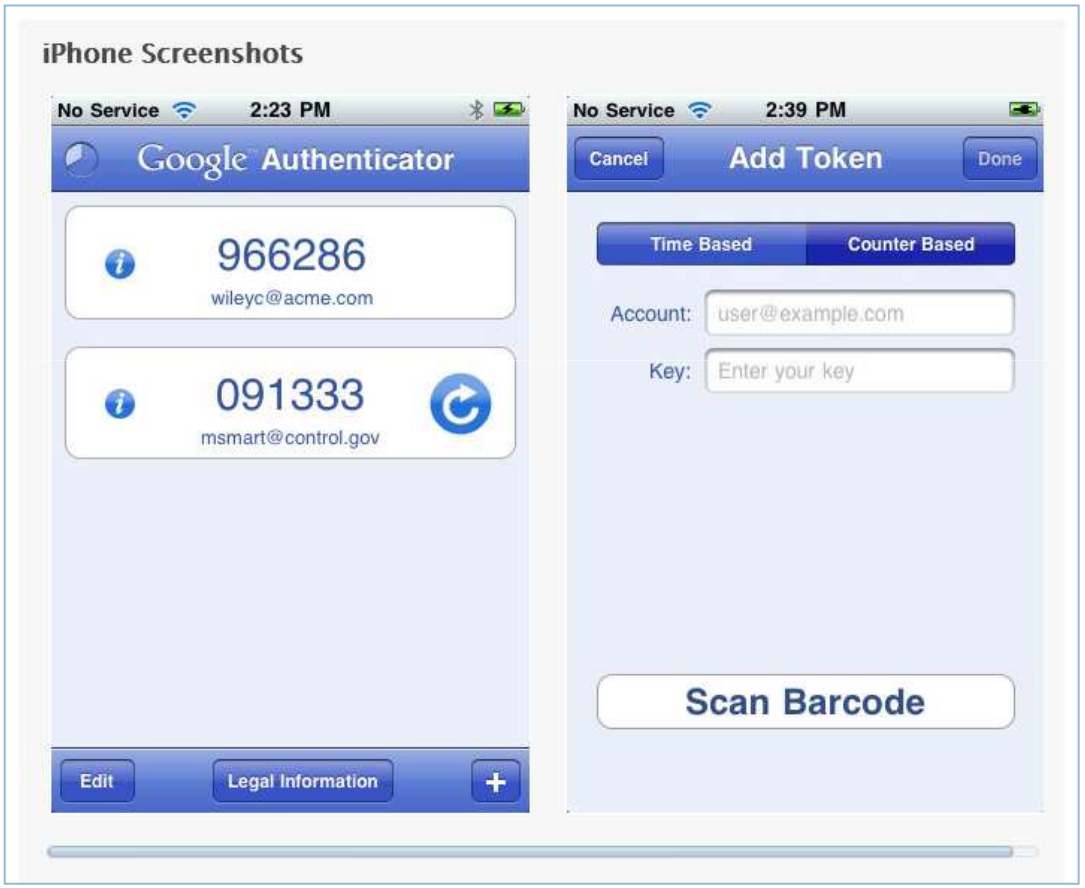
http://itunes.apple.com/us/app/iotp/id328973960

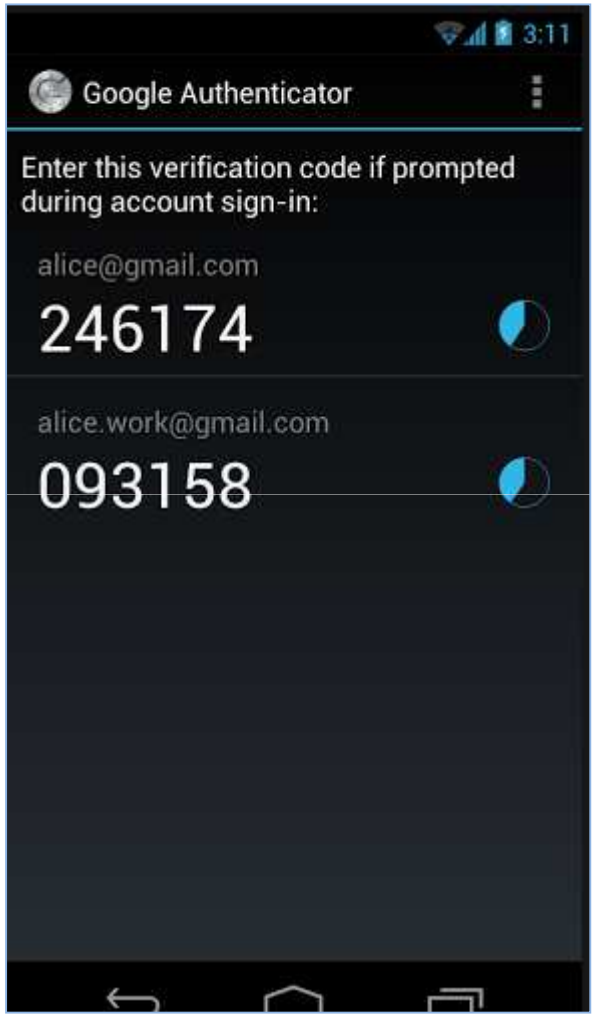
google-authenticator

- These implementations support
  - HMAC-Based One-time Password (HOTP) algorithm specified in RFC 4226
  - Time-based One-time Password (TOTP) algorithm specified in RFC 6238
  - Google Authenticator
    - Android, IOS and Blackberry

http://code.google.com/p/google-authenticator/

google-authenticator



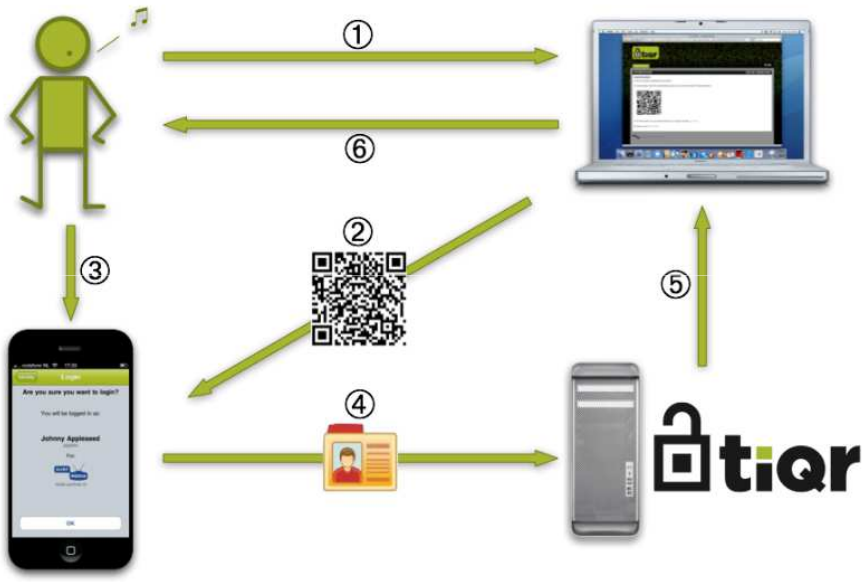


OCRA on a mobile

OCRA on Mobile

Data flow

The diagram below shows the data flow for a normal, successful authentication using tiqr:



The diagram shows the following steps:

1. The user surfs to a web site and is required to log in
2. The web site displays a QR code
3. The user scans the QR code using the tiqr App on his or her phone, confirms login and enters his/her PIN
4. The user's identity together with the response to the challenge encoded in the QR tag is sent to the server using the phone's Internet connection
5. The server validates the response and authorises login
6. The browser reloads the page and the user is logged in

OTP without PIN

OTP Pin Protected

OTP on Smartcard

OTP with Smartcard

OTP hybrid (OTP & PKI)

YubiKey

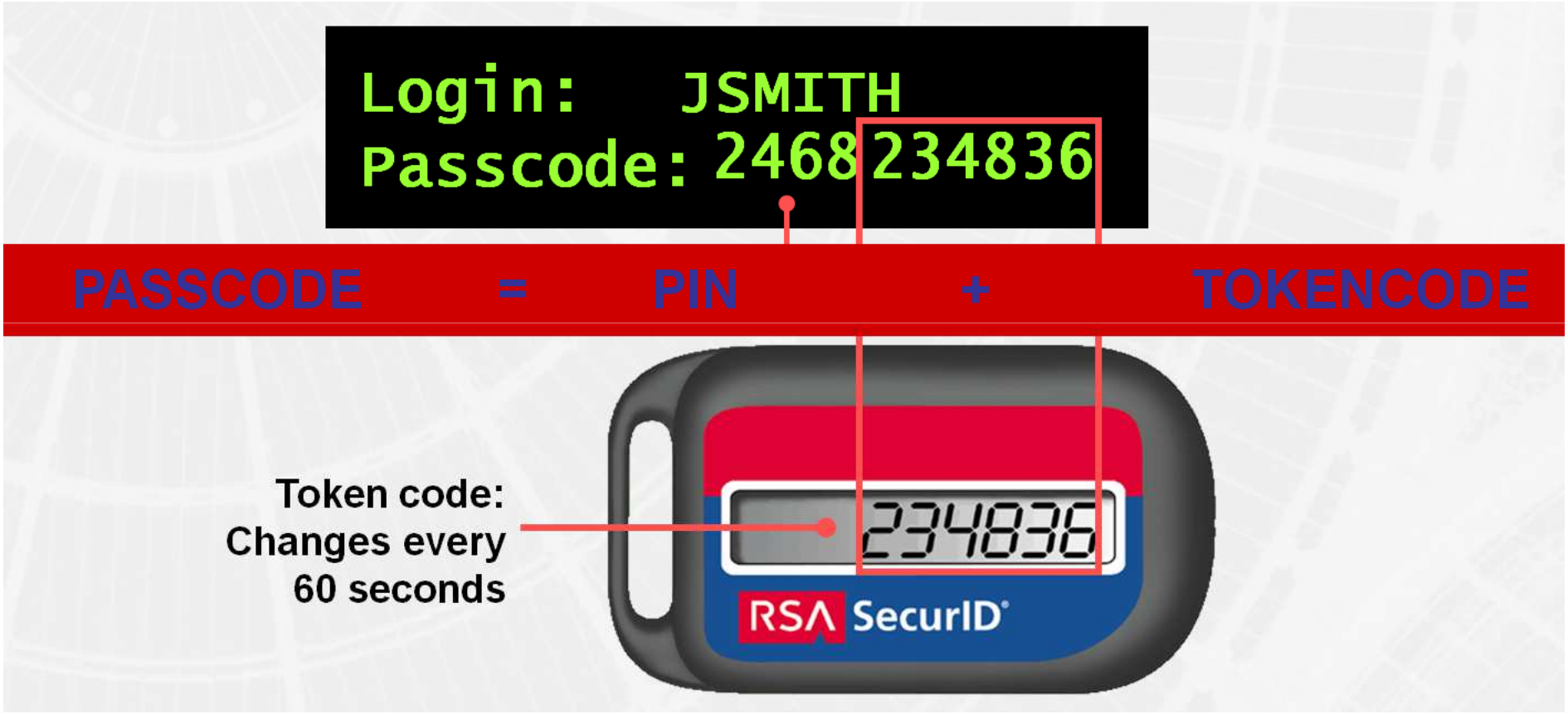
YubiKey

Feature	YubiKey 1	YubiKey 2
		
Introduced	2008	2009
Weight	1.8 g (0.06 oz)	3.3 g (0.12 oz)
Dimensions	45 x 18 x 2.3 mm (1.8 x 0.7 x 0.1 inch)	45x 18 x 3 mm (1.8 x 0.7 x 0.12 inch)
Color	Black only	Black and White standard. Others on request.
USB	2.0 Low-speed	2.0 Low-speed
Configurations	1	2
Static password mode	Basic from firmware revision 1.3	Enhanced
OATH-HOTP	No	From firmware revision 2.1
Challenge-response mode	No	From firmware revision 2.2
Password update by user	No	Yes
Construction	Two piece + resin	Mono-block mold, hermetical
Protection class (non-certified)	IP 51	IP 67
Max bending force	5 N	25 N
EMC	CE 89/336/EEC FCC 47 CFR Part 15	CE 89/336/EEC FCC 47 CFR Part 15

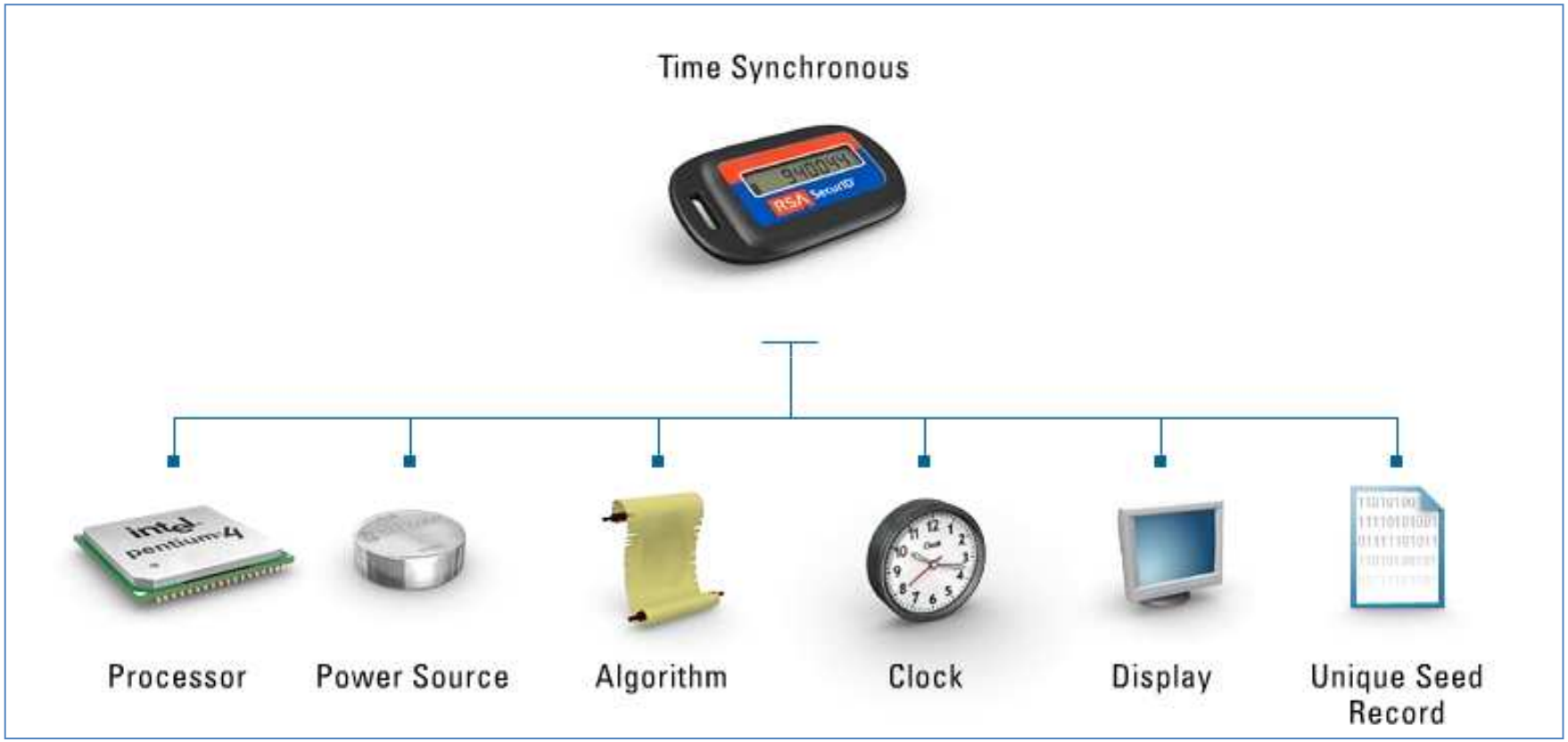
Yubikey

- http://www.yubico.com/support/documentation/
- http://forum.yubico.com/
- http://code.google.com/p/yubico-pam/

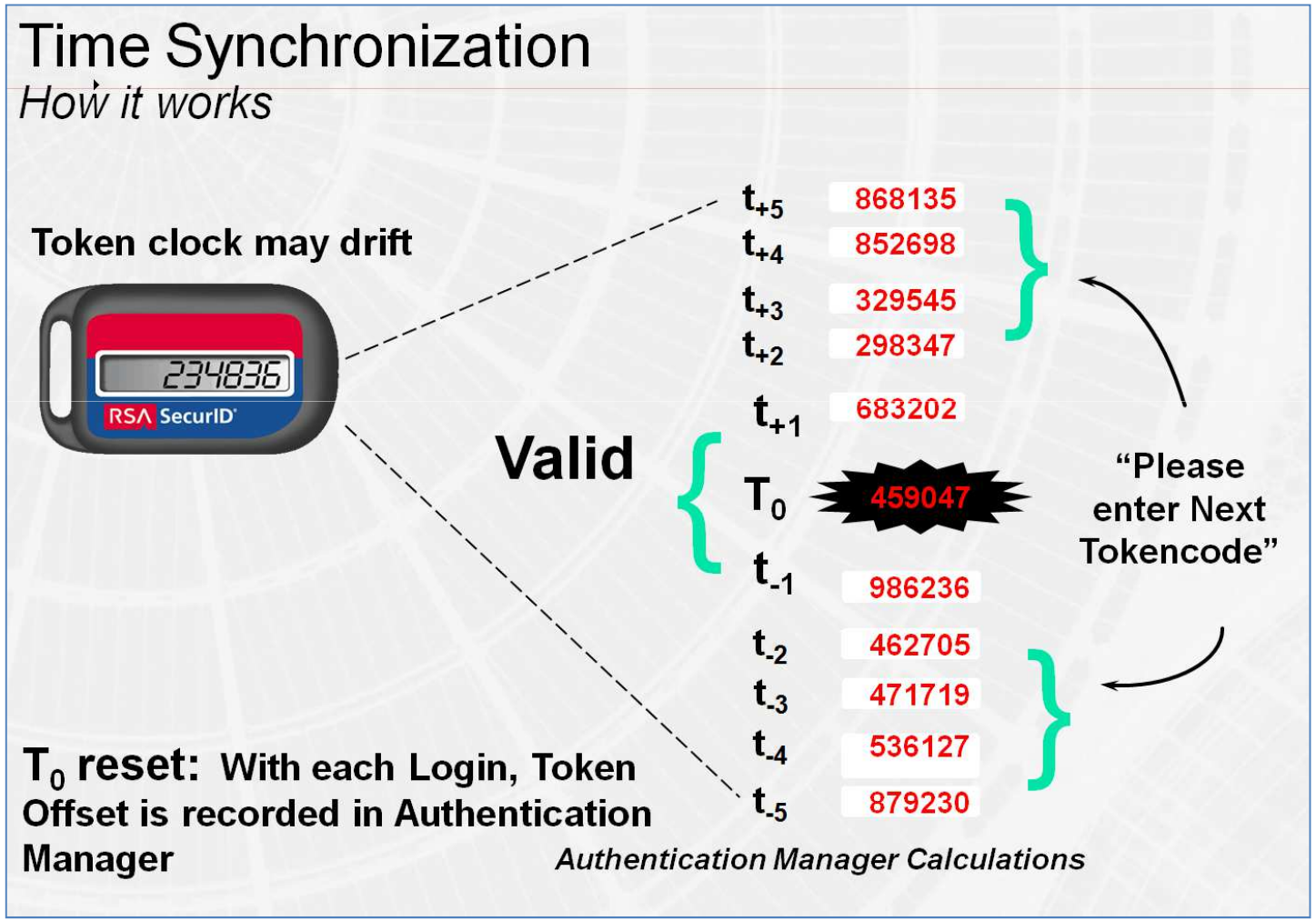
RSA SecurID 1/3



RSA SecurID 2/3



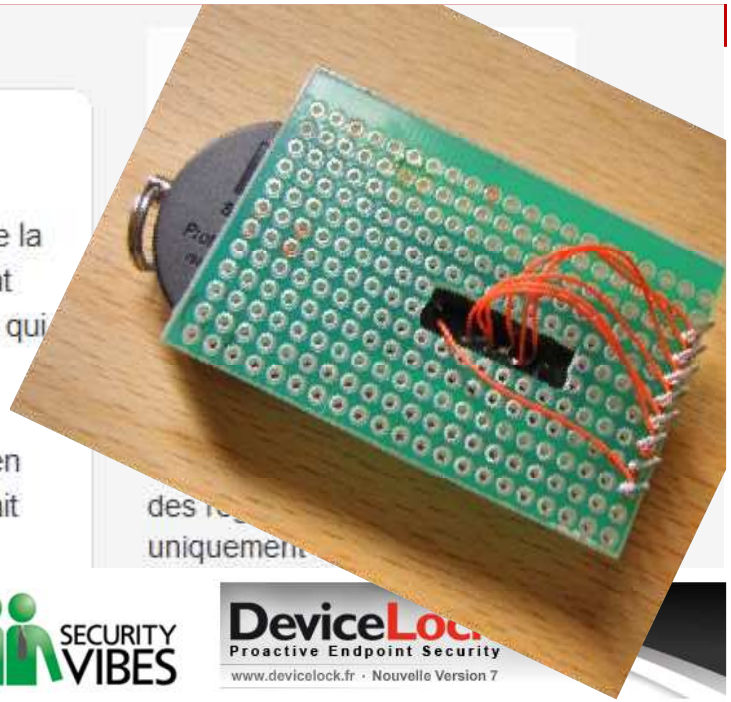
RSA SecurID 3/3



MERCREDI, 30 MARS 2011

Hack RSA SecurID : l'histoire n'est pas terminée !

Beaucoup de personnes dans la communauté de la sécurité informatique se demandent pourquoi tant de silence de la part de RSA SecurID. La phrase qui ressort le plus est : « RSA Silent About Compromise For 7 Days – Assume SecurID Is Broken ». Effectivement, le doute n'est pas bon en sécurité des systèmes d'information. Mais que fait donc RSA ?



Selon moi, comme mentionné dans mon 1er billet, cette histoire est un évèn
marquant de la sécurité informatique.

SECURITY VIBES

DeviceLoc
Proactive Endpoint Security
www.devicelock.fr · Nouvelle Version 7

ICTTF - International Cyber Threat Task Force

Home ICTTF Blogs Chat Docs Events Forum Groups Likes Lists News F

Paul C Dwyer

View All Entries
Share
Report

Search Entries

RSA SecurID Hack - My Opinion

Posted by Paul C Dwyer March 26 - Filed in Authentication - #APT #rsa #securid - 1,676 views

4

Recommend

Okay so what's all this about the RSA SecurID Hack?

Since the news, I have received a large number of calls and emails for my opinion
So here it is, hope it helps and as always it's my personal opinion and not that of

So let's look at some background and key facts. The RSA SecurID technology is
for authentication.

Mostly used for corporate and government VPN's it is the de facto way of authenti

Alertes & Menaces | Business | Technologies | Conformité & organisation | Carri

ACCUEIL » ALERTES & MENACES »

Hack RSA : la suite de l'histoire

Jérôme Saiz le 30 mars 2011 11 h 21 min dans la rubrique Alertes & Menaces / 5 Commentaires, rejoignez la discussion! et 3 réactions

featured · hack · jetons · RSA · secureid · seed · seeds · token · token rsa

Télécharger l'article au format PDF

Tweet 4 | Buzz 2 | Share 3 | Like

Une mise au point essentielle, recueillie par un membre de *SecurityVibes*, éclaire un peu mieux les conséquences du piratage dont à récemment été victime *RSA*, la division sécurité d'EMC, si la clé secrète était effectivement volée. Selon ses informations, trois éléments entrent en compte dans une clé *RSA* :

- La clé secrète (volée)
- Le numéro de série du *token* (9 chiffres)
- L'heure qu'il est (connue de tout le monde)

D'après notre membre la clé secrète serait volée (ce que *RSA* n'a cependant pas confirmé *officiellement*), et il ne

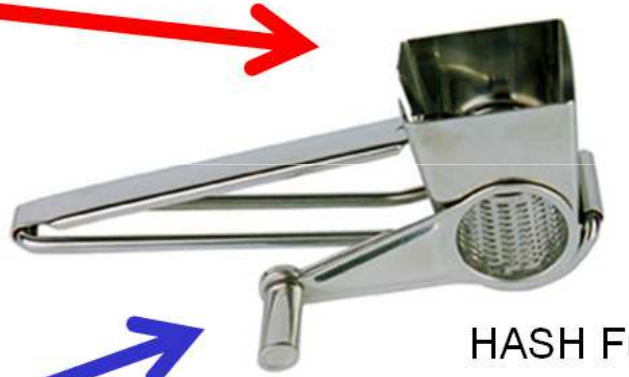
NOTEZ L'ARTICLE!

Where are[is] the seed ?

Crypto-101 / Time Based OTP

S=Serial Number ?

K=Secret Key / Seed



Tokencode

T=UTC Time

HASH Function
RSA SecurID

ie = OTP(K,T,S) = Truncate(HASH(K,T,S))

Seed generation & distribution ? Still a good model ?

Threat Agent (APT)

K1

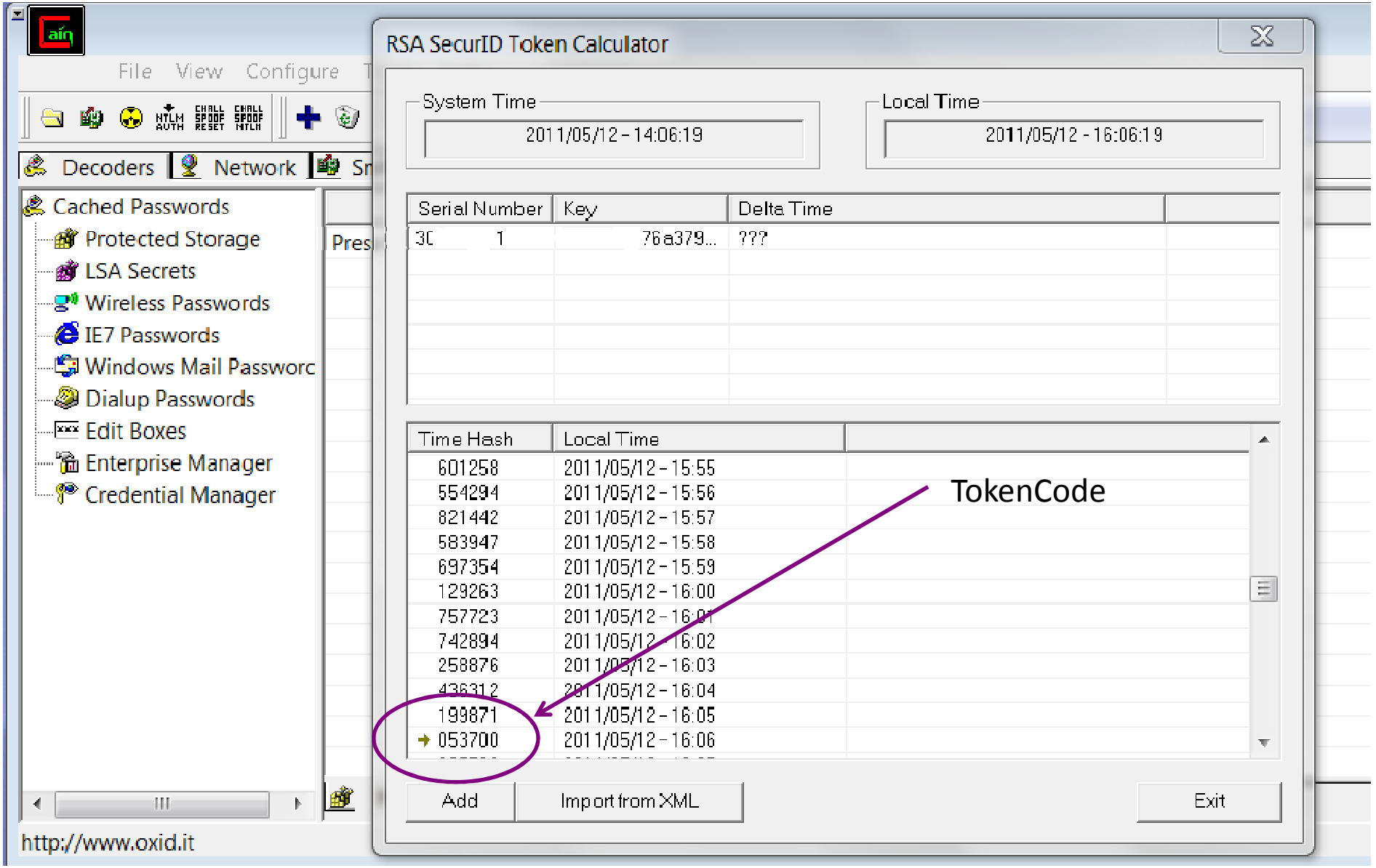
Editor / Vendor

Secret Key are[is] generated on premise

K1

K1

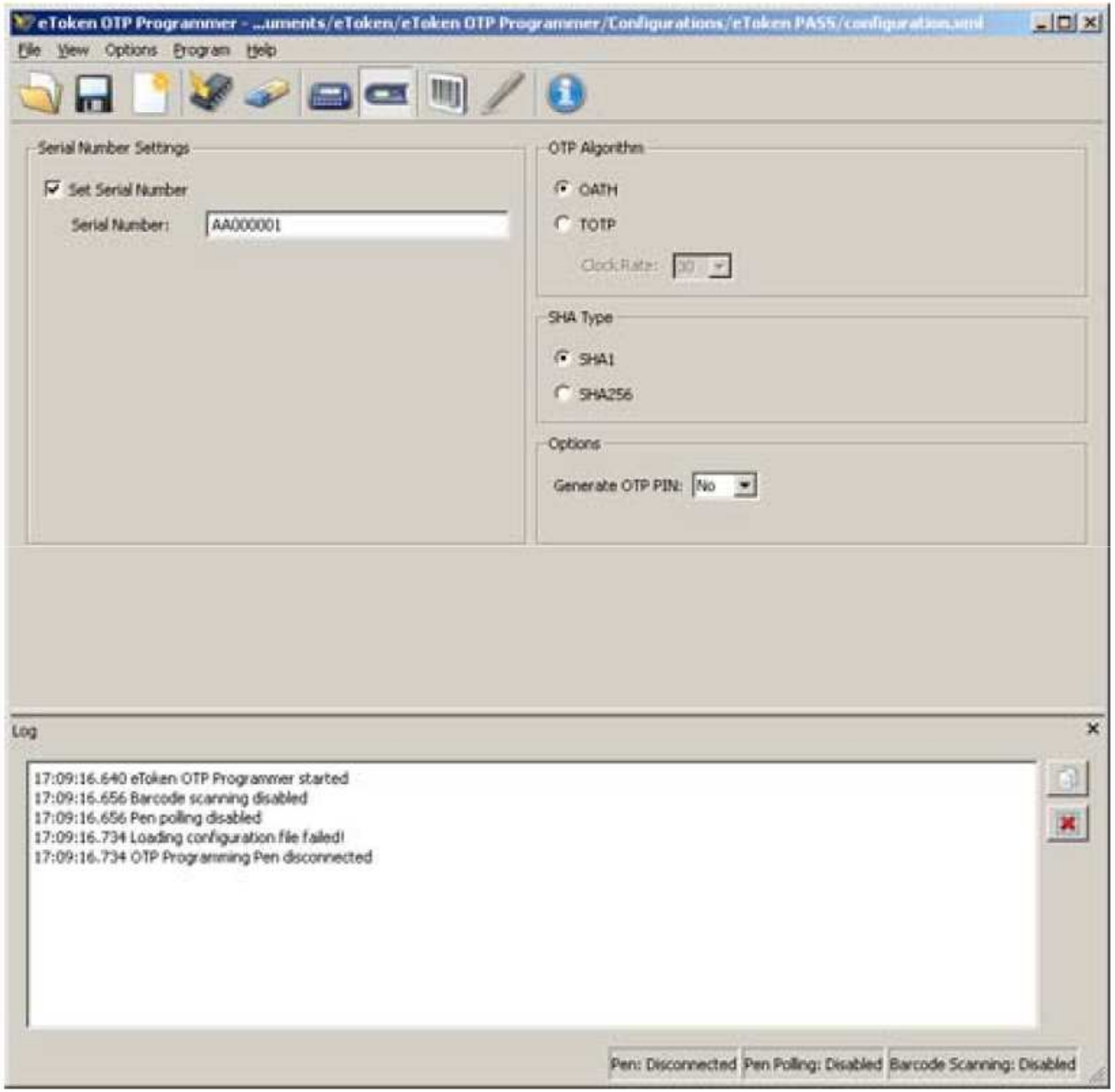
RSA SecurID


File View Configure
Decoders Network
Cached Passwords
Protected Storage
LSA Secrets
Wireless Passwords
IE7 Passwords
Windows Mail Passworc
Dialup Passwords
Edit Boxes
Enterprise Manager
Credential Manager
http://www.oxid.it
RSA SecurID Token Calculator
System Time
2011/05/12 - 14:06:19
Local Time
2011/05/12 - 16:06:19
Serial Number
Key
Delta Time
Time Hash
Local Time
601258 2011/05/12 - 15:55
554294 2011/05/12 - 15:56
821442 2011/05/12 - 15:57
583947 2011/05/12 - 15:58
697354 2011/05/12 - 15:59
129263 2011/05/12 - 16:00
742894 2011/05/12 - 16:02
258876 2011/05/12 - 16:03
199871 2011/05/12 - 16:05
053700 2011/05/12 - 16:06
TokenCode
Add
Import from XML
Exit

Generate Seed on premise

Features

- OTP authentication device with LCD display, battery, and OTP generation button
- Time-sync and Event-sync options
- Field programmable
- Support for OATH TOTP protocol
- Standard support for RADIUS OTP
- Modular OTP algorithm support

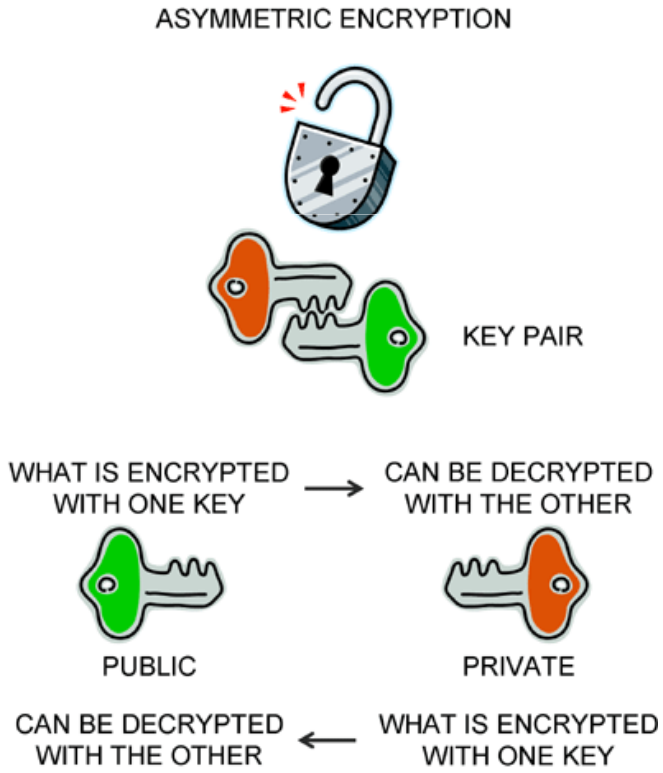

eToken OTP Programmer - ...uments/eToken/eToken OTP Programmer/Configurations/eToken PASS/configuration.xml
File View Options Program Help
Serial Number Settings
Set Serial Number
Serial Number: AA000001
OTP Algorithm
OATH
TOTP
Clock Rate: 30
SHA Type
SHA1
SHA256
Options
Generate OTP PIN: No
Log
17:09:16.640 eToken OTP Programmer started
17:09:16.656 Barcode scanning disabled
17:09:16.656 Pen polling disabled
17:09:16.734 Loading configuration file failed!
17:09:16.734 OTP Programming Pen disconnected
Pen: Disconnected
Pen Polling: Disabled
Barcode Scanning: Disabled

PKI

PKI AuthN

PKI AuthN

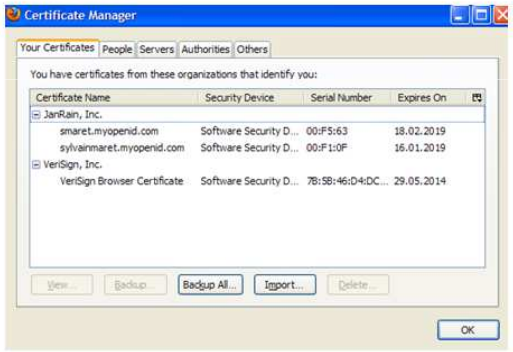
- Based on asymmetric encryption



PKI Tokens Storage

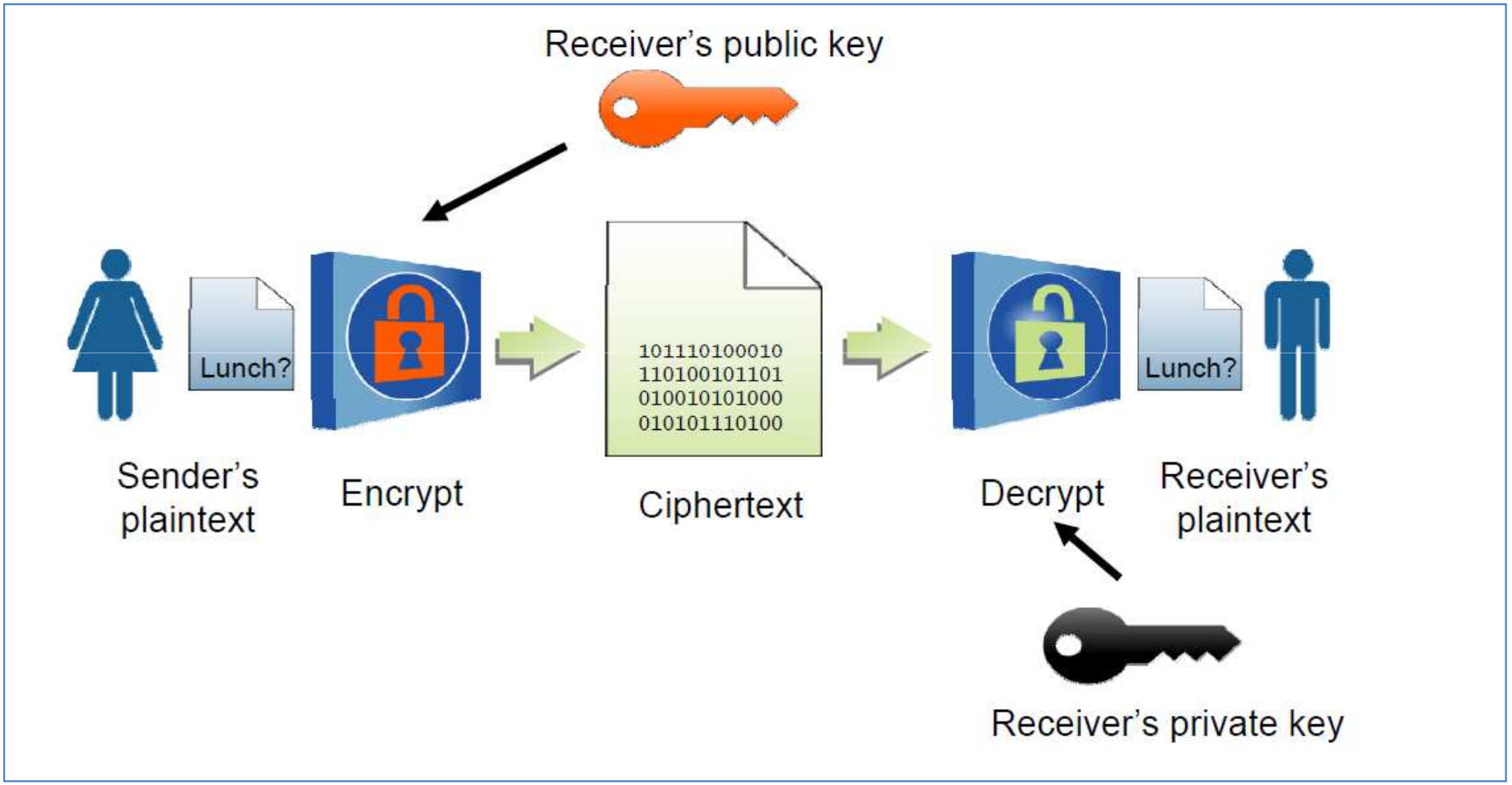
PKI: Digital Certificate

Software Certificate (PKCS#12;PFX)

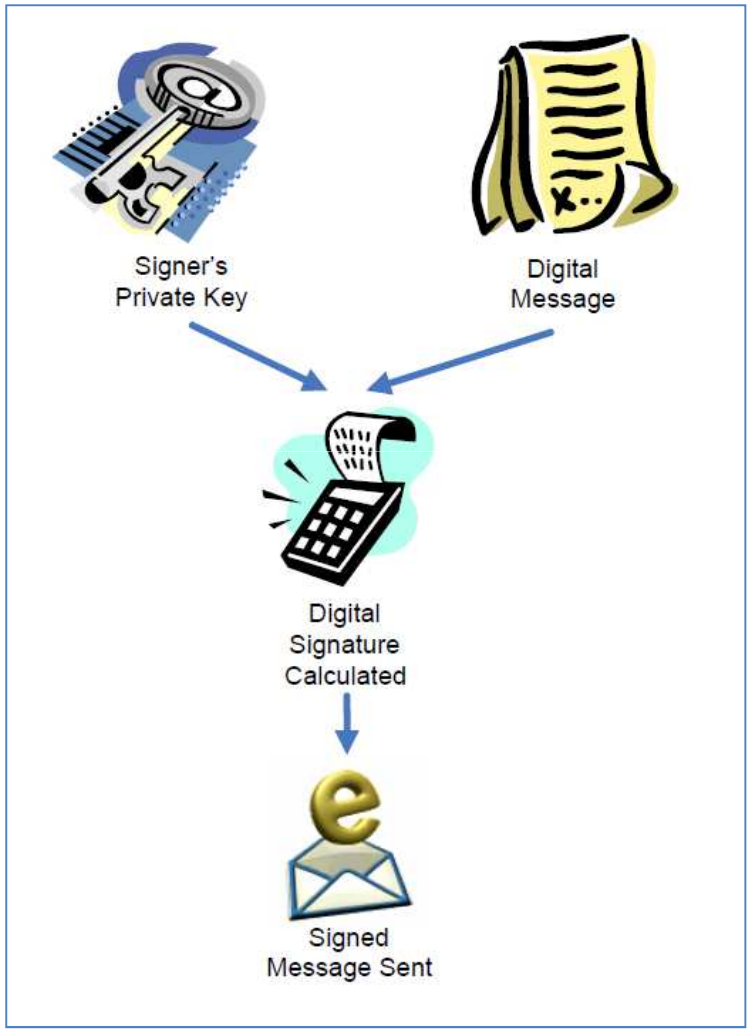


Hardware Token (Crypto PKI) Strong Authentication

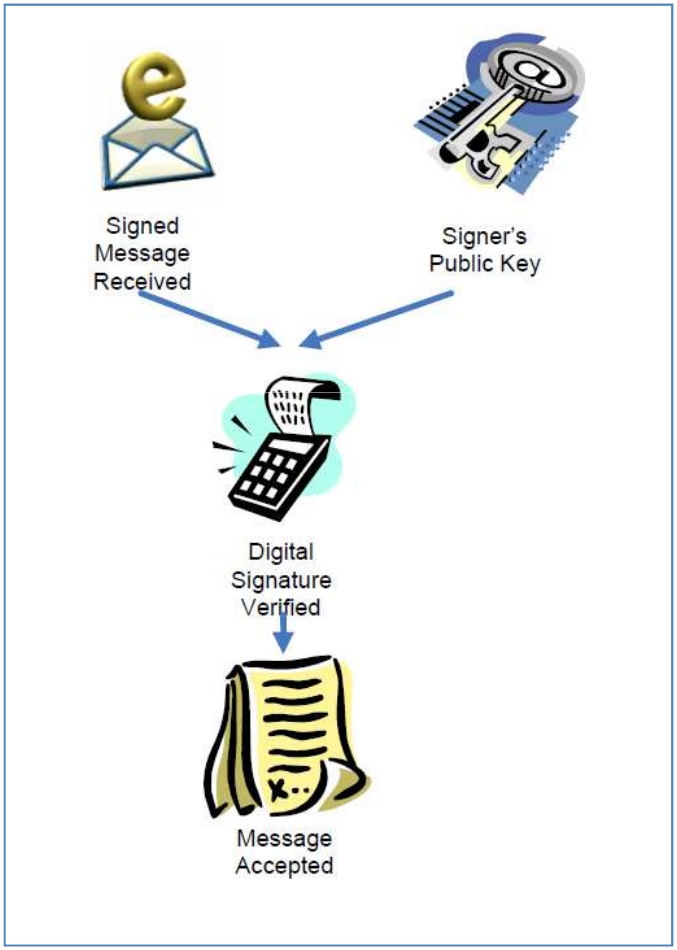
Public Key Cryptography 101



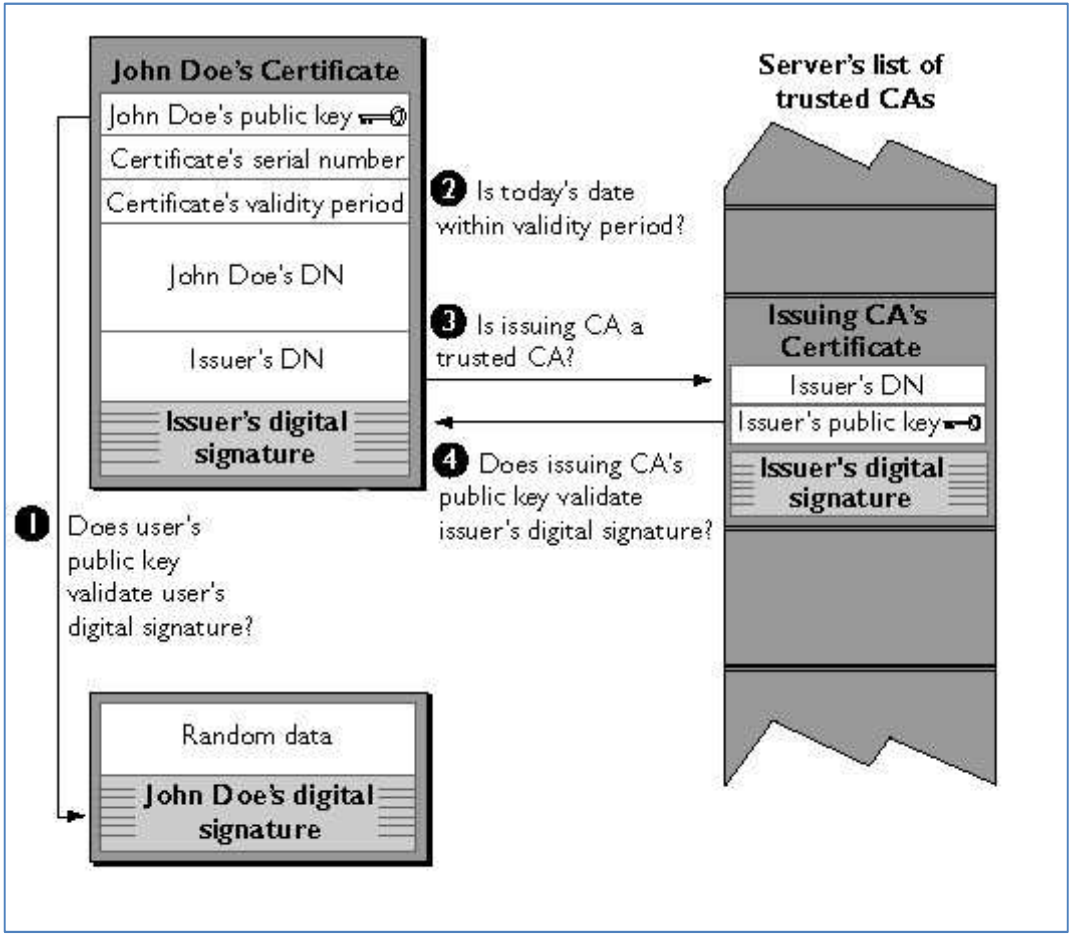
Signature 101



Signature – Verification 101



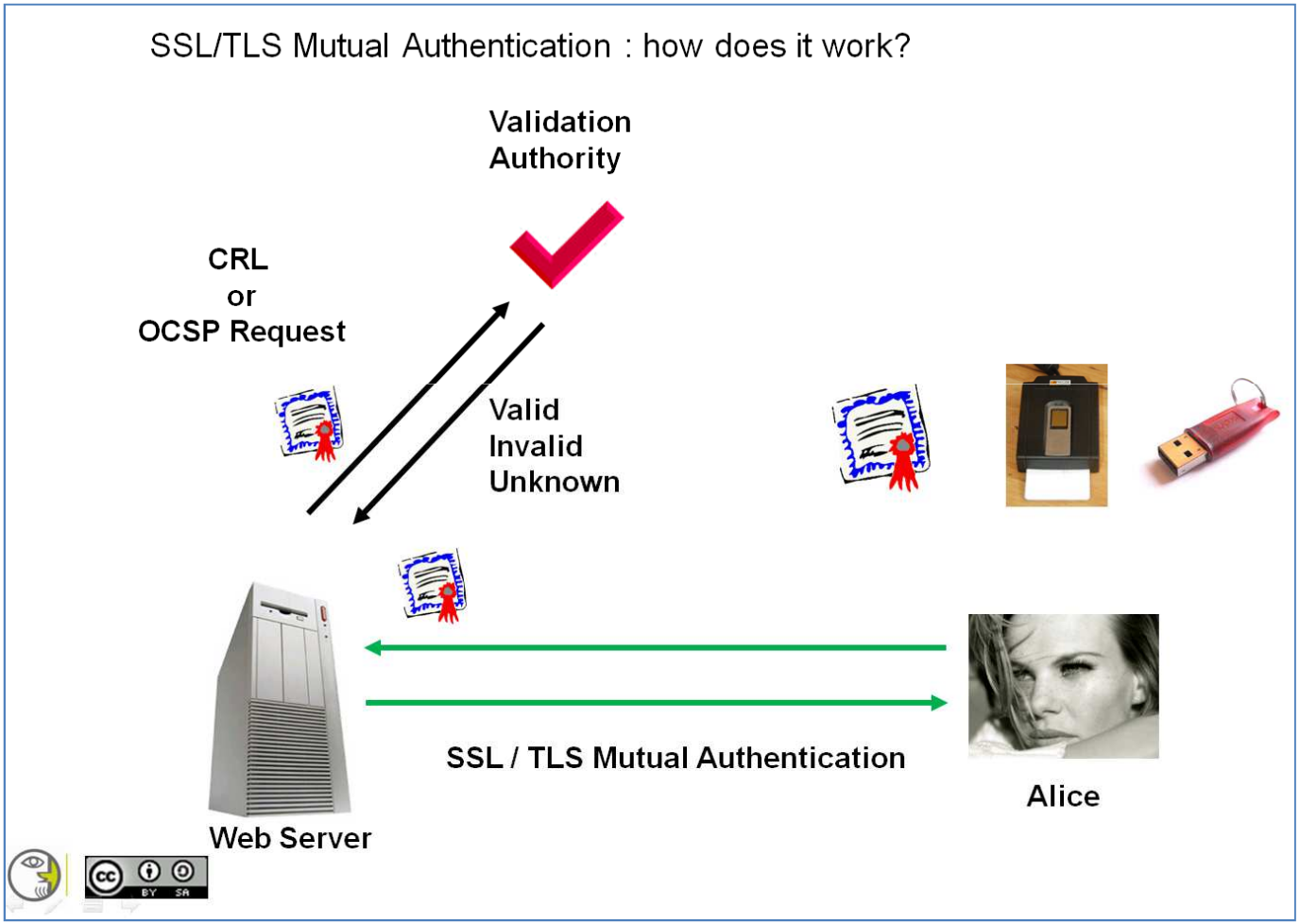
Mutual AuthN SSL



PKI Certificate Validation

- CRL
- Delta CRL
- OCSP

OCSP Validation

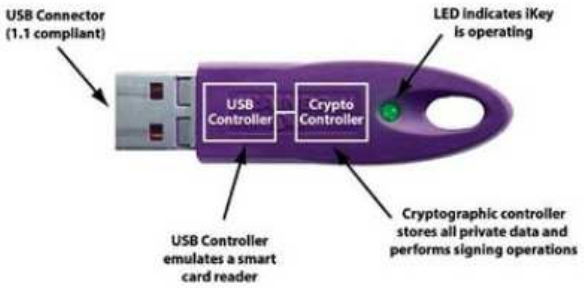


Physical and Logical Components

Types of cards and tokens

- Card
  - Plastic with embedded microchip
  - Credit card form factor
  - More real estate
  - Massive production capability because of bank card market
  - Good "wallet appeal"

- USB Token
  - Same chip stored in USB device
  - Often looped for "keychain dongle"
  - No reader required
  - Easy to embed antenna
  - Good size to add biometric
  - But lack of real estate prevents badging



Physical and Logical Components

Smart cards are not…

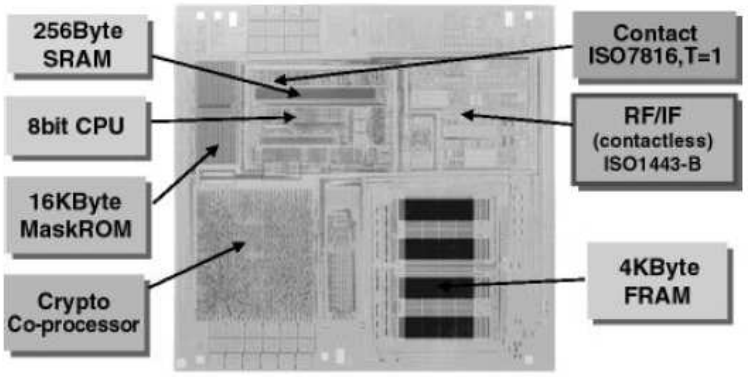
- Magnetic stripe cards
  - Like most bank debit/credit cards
  - Loyalty cards
  - All processing is performed off-card
  - Stripe encodes small amount of data
  - (A smart card can add a mag stripe)
- Memory cards
  - Flash (or other) RAM for stored values onboard
  - Processing performed off-card
  - Fairly large memory sizes available today (USB drives)
  - (Smart cards contain memory, but it's secondary to the CPU)

Physical and Logical Components

burton GROUP

The chip

- Interfaces
  - Contact (ISO 7816)
  - Contactless (ISO 14443)
- Processing
  - General microprocessor unit (MPU/CPU)
  - Cryptographic operations
  - Random number generator
- Memory
  - Flash for persistent
- Countermeasures
  - Anti-DPA
  - Electrical protection



Source: Fujitsu

Crypto Processor

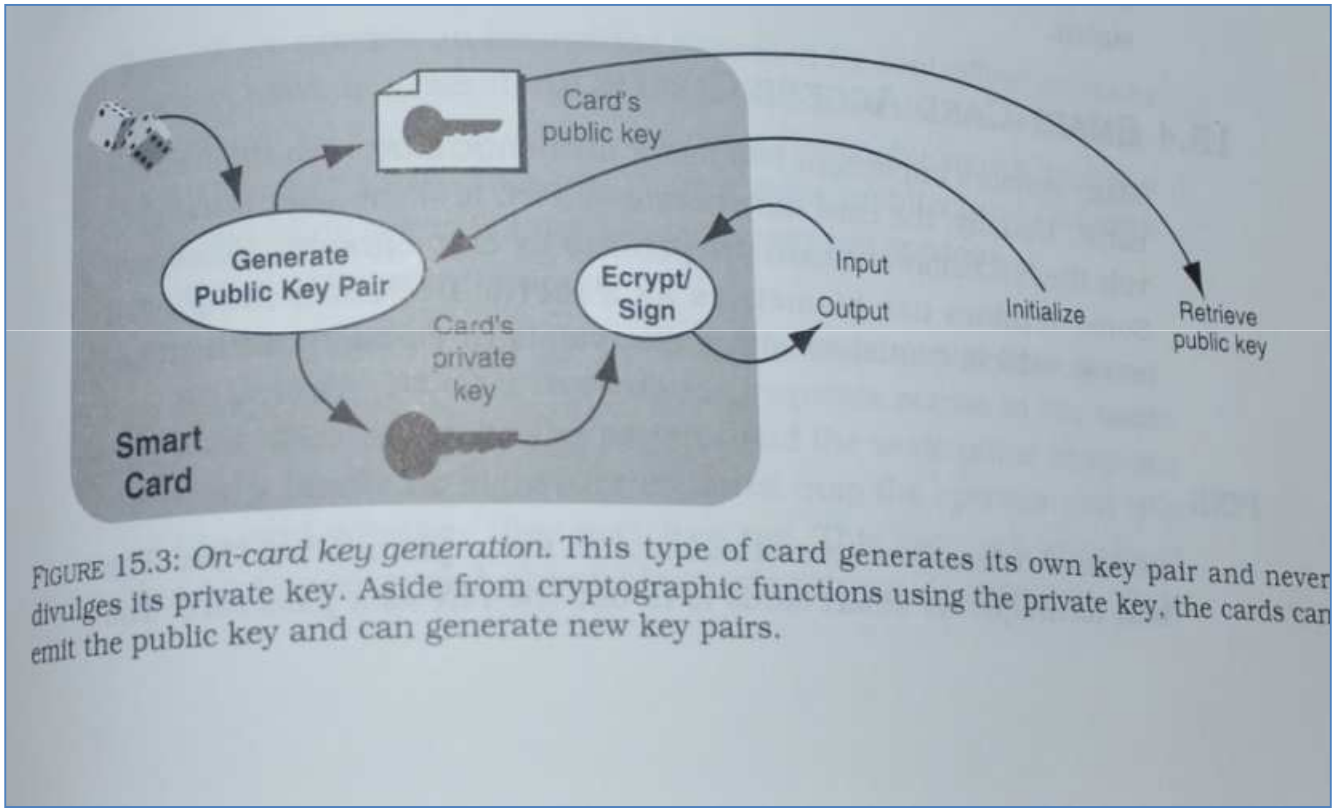


FIGURE 15.3: *On-card key generation.* This type of card generates its own key pair and never divulges its private key. Aside from cryptographic functions using the private key, the cards can emit the public key and can generate new key pairs.

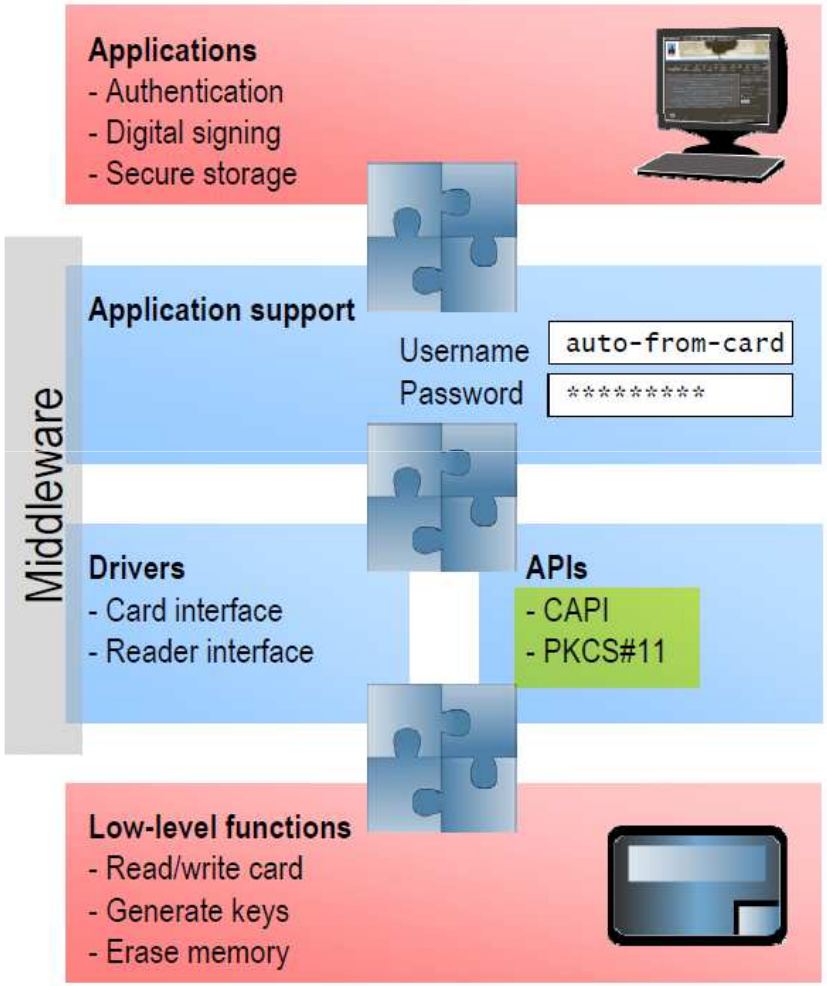
Source: Richard E. Smith / Authentication

Physical and Logical Components

burton GROUP

Middleware

- Driver software
  - PC/SC
  - CAPI CSP & PKCS#11
- APIs for application support
  - Cryptographic functions
  - Data storage
  - Card management
- Special-purpose features
  - Password management
  - Single sign-on



Standards

Programming

- PC/SC API
  - winscard.dll on Windows; OpenSC/MUSCLE on open source
  - Low level abstraction for talking to reader
- Windows cryptography API (CryptoAPI or CAPI)
  - Programming interface for cryptographic functions in Windows
  - Card vendors provide cryptographic service provider (CSP) that implements the API
  - Example calls: `CryptHashData()`, `CryptGenKey()`, `CryptEncrypt()`
- PKCS#11 (also called cryptoki)
  - "Cryptographic Token Interface Standard "
  - Similar concept to CAPI for cross-platform environments
  - Developed by RSA
  - Used by Netscape, Mozilla, and other security applications
  - Example calls: `C_Digest()`, `C_GenerateKey()`, `C_Encrypt()`

Smart Card

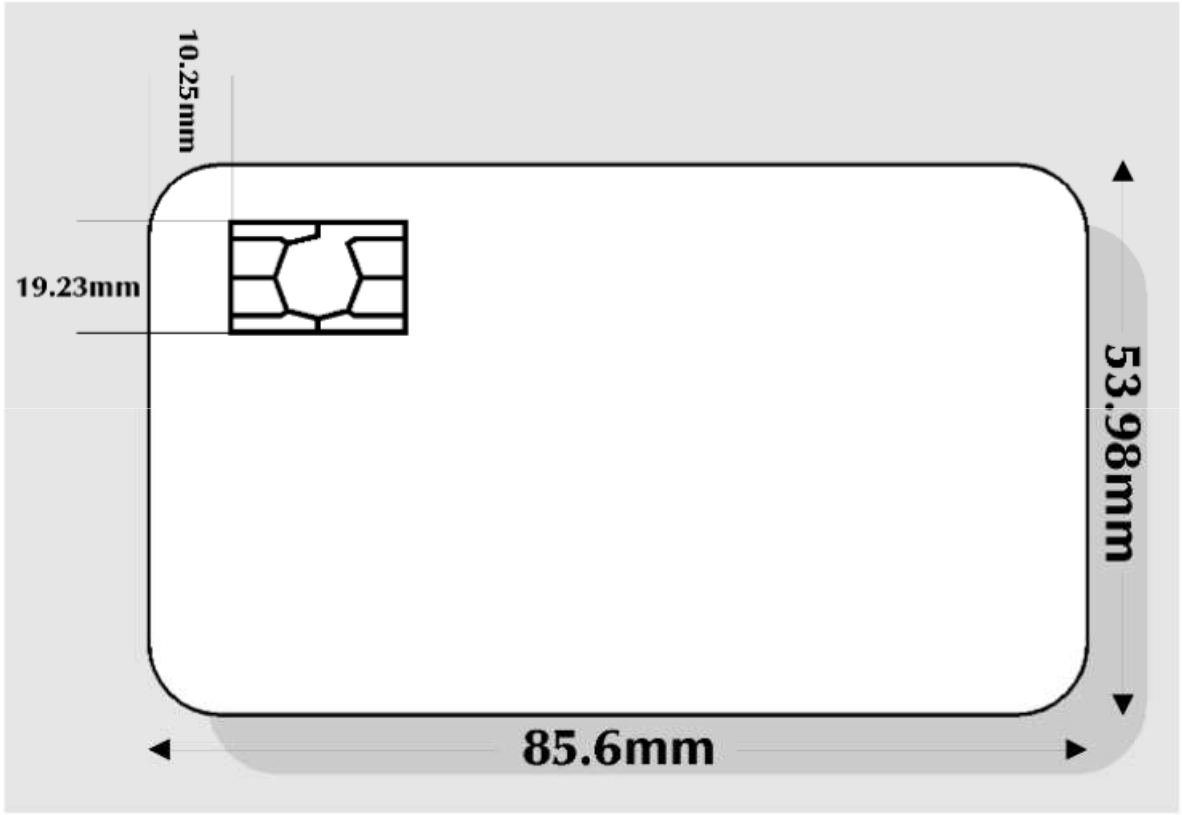


Figure 1: ISO ID 1Card

Smart Card

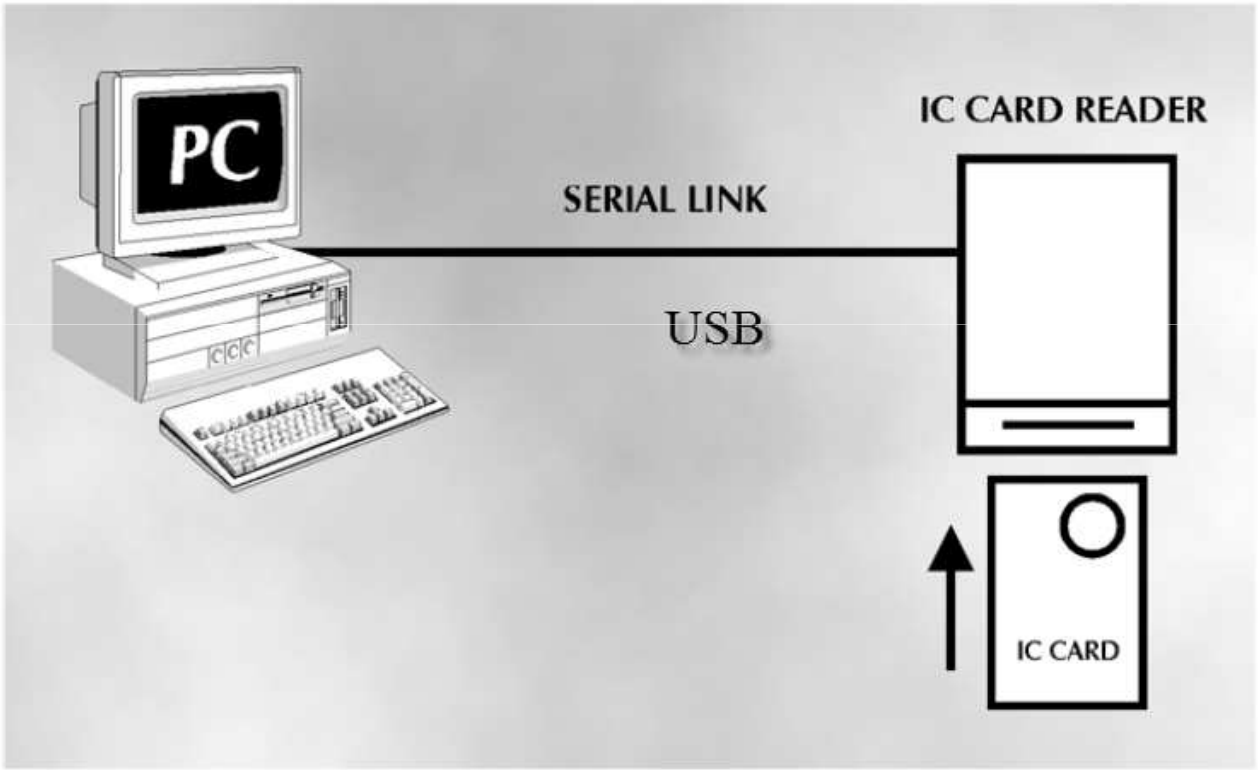


Figure 2: Basic Card Evaluation Configuration

Smart Card - Crypto

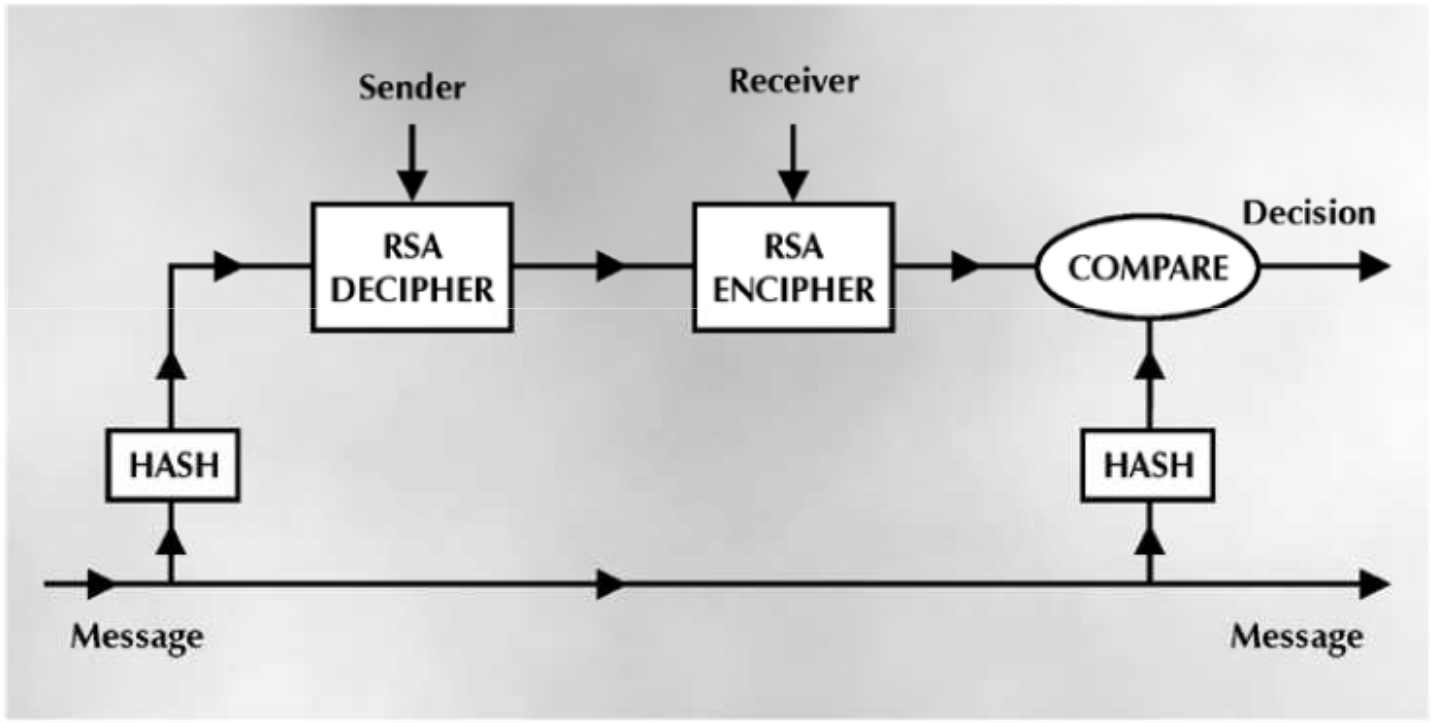


Figure 1: A RSA Digital Signature

Standards

Key and file system management

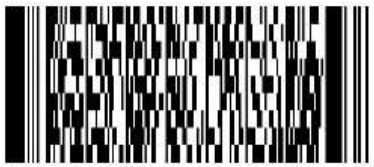
- PKCS#12
  - "Personal Information Exchange Syntax Standard"
  - Another part of RSA's public key cryptography standards
  - Secure transport of private keys, certificates, and other secrets
  - Typically passphrase-protected
  - Often used to transfer keys from card to card
- PKCS#15
  - "Cryptographic Token Information Format Standard"
  - Interoperable format for placing information on cards
  - Independent of the access method (PKCS#11, CAPI, etc.)
  - Standardizes management of applications, keys, certificates, and other data on cards

Physical and Logical Components

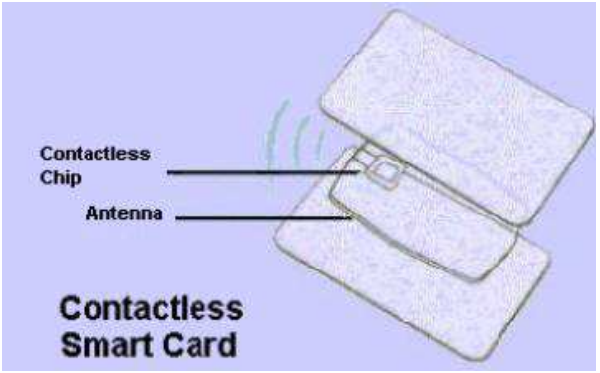
burton GROUP

SmartBadge

- User picture
- Company logo
- 1D barcode
  - ID number
- 2D barcode
  - Biometric or other information
- Magnetic stripe
- Wireless/proximity antenna
- Smart chip

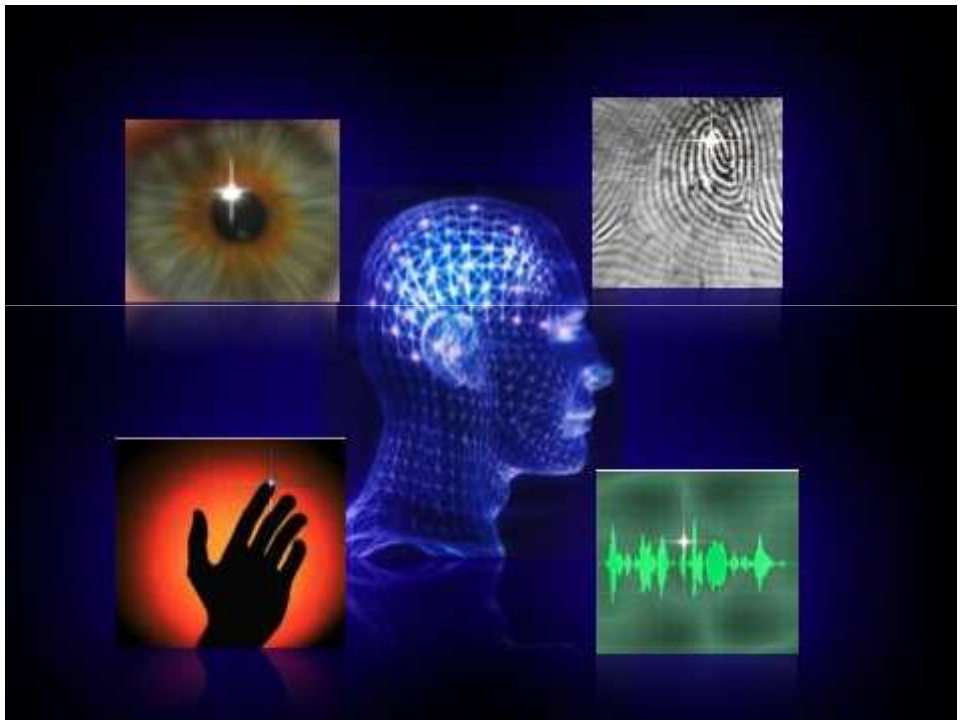


PKI Tokens

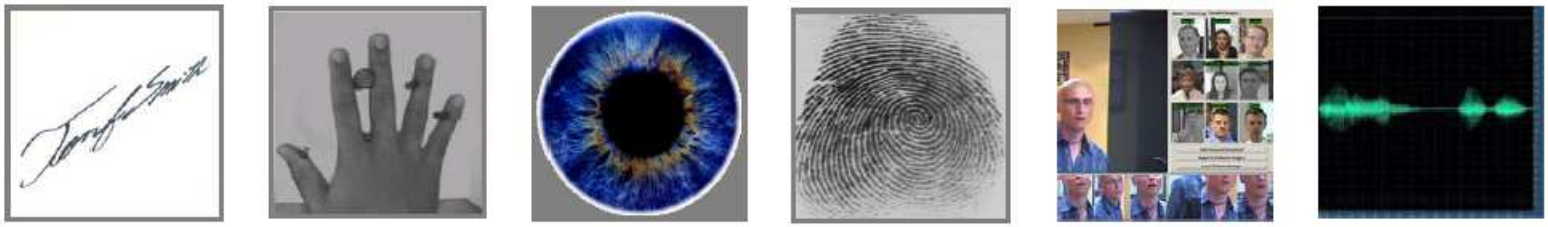


Biometrics

BIO AuthN



Biometrics



General term used alternatively to describe a characteristic or a process

As a Characteristic it is a measurable biological (anatomical and physiological) and behavioral characteristic that can be used for automated recognition

As a Process it encompasses automated methods of recognizing an individual based on measurable biological (anatomical and physiological) and behavioral characteristics

Source: http://www.biometrics.gov/

Biometric Terms

Verification occurs when the biometric system attempts to confirm an individual's claimed identity by comparing a submitted sample to one or more previously enrolled templates

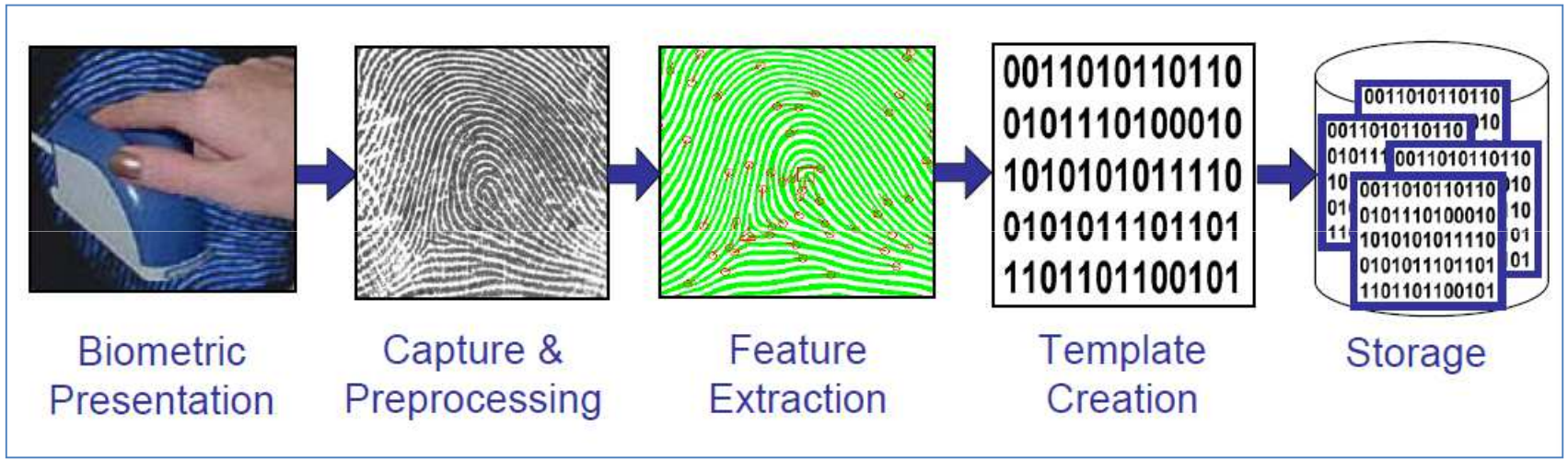
Identification the biometric system searches a database for a reference matching a submitted biometric sample, and if found, returns a corresponding identity. A biometric is collected and compared to all the references in a database.

- Identification is "closed-set" if the person is known to exist in the database
- In "open-set" identification, the person is not guaranteed to exist in the database. The system must determine if the person is in the database

Recognition is a generic term and does not necessarily imply either verification or identification. All biometric systems perform "recognition"

Source: http://www.biometrics.gov/

Enrollment Process



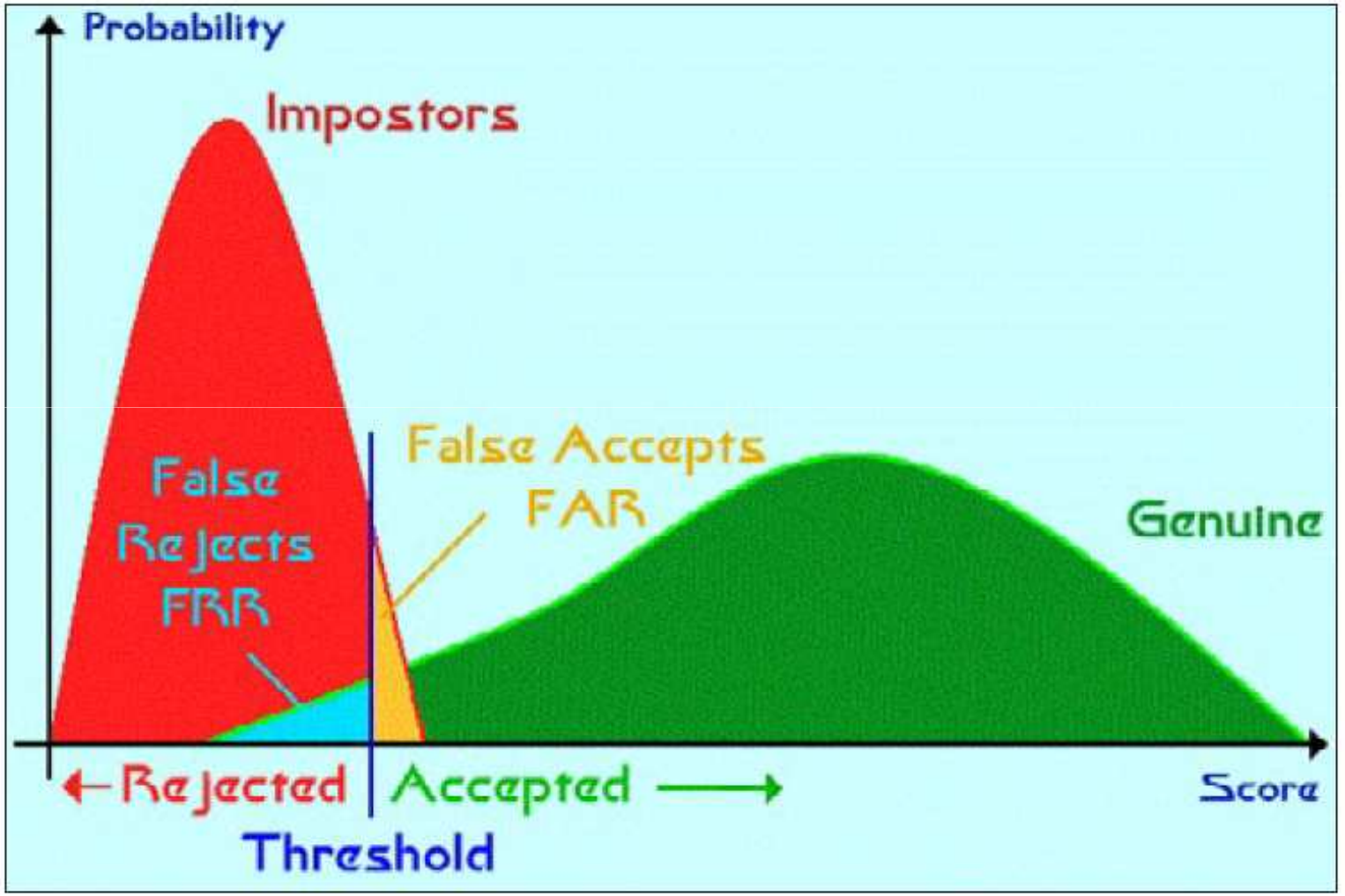
Source: http://www.biometrics.gov/

Components

- **Sensor** - collects data and converts the information to a digital format
- **Signal processing algorithms** - perform quality control activities and develop the biometric template
- **Data storage** - keeps information that new biometric templates will be compared to
- **Matching algorithm** - compares the new biometric template to one or more templates in data storage
- **Decision process** - uses the results from the matching component to make a system-level decision (either automated or human-assisted)

Source: http://www.biometrics.gov/

FRR / FAR



Source: http://www.biometrics.gov/

TAR

True Accept Rate

- End user must first make a claim as to his/her identity (e.g., I am John Q. Public)
- Biometric system then determines if the end-user's identity claim is true or false
- The gentleman at bottom makes a claim that he is the gentleman at top. Assume that the system's verification threshold was set at 0.90
  - Since 0.93 is higher than 0.90, the system in this example has correctly determined that the gentleman in the top picture is the same as the gentleman in the bottom picture
  - This is called a true accept or correct verification
- Now assume that the same individual makes the same claim, except the system's verification threshold is 0.95. The demonstration face recognition system will not make a correct decision
- After many trials with this gentleman, as well as other correct matches, we will know the rate legitimate end users are correctly verified by the system. This is called the true accept or correct verification rate

Source: http://www.biometrics.gov/

FAR

False Accept Rate

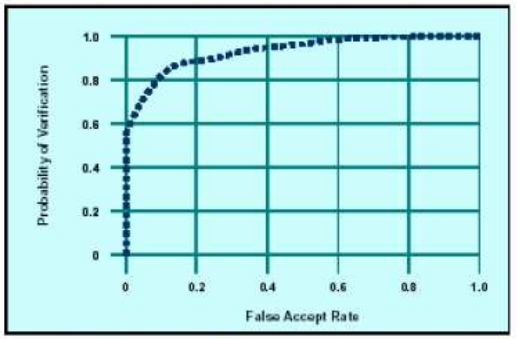
- The gentleman on the bottom claims to be the gentleman at top
  - Assume that the system returns a similarity score of 0.86 and verification threshold was set at 0.9
  - Face recognition system determines the gentleman on the bottom is not the gentleman on the top
- Look at the case where the same individual makes the same claim, but the system's verification threshold is set at 0.85
  - The system incorrectly verifies that the gentleman is the gentleman in the system
  - This error is called a false accept
  - Trials run with incorrect claims will determine the rate at which the system incorrectly matches an imposter individual to another individual's existing biometric

Ideally, biometric systems would always provide a probability of verification of 100% with a false accept rate of 0%

Source: http://www.biometrics.gov/

Accept Rate Threshold

- Optimal system threshold for a given application
- Determining threshold can be difficult because the verification rate and false accept rate are not independent variables
- If the threshold in the example face recognition system is raised, the verification rate decreases, but the false accept rate also decreases
- If the threshold in the example system is lowered, the verification rate increases, but the false accept rate also increases
- Plotting verification accept rates against the associated false accept rates, called a Receiver Operating Characteristic (ROC) curve, allows for a visualization of this trade-off relationship
- Varying the system's threshold moves the operating point along its ROC curve



Source: http://www.biometrics.gov/

Identification

Open-set identification, the biometric system determines if the individual's biometric template matches a biometric template of someone in the database

Task examples:

- Comparing biometrics of visitors against a terrorist database
- Comparing a biometric of a "John Doe" to a missing person's database

Face recognition system:

- The system first compares the submitted image to each image in the database. Assume that the similarity score for each comparison is 0.6, 0.86, 0.9, and 0.4, respectively. Also assume that the system's watchlist threshold is set at 0.85
- Face recognition system sounds an alarm each time one or more of the similarity scores is higher than the threshold
- Since an alarm sounded, the system user would look more closely at the similarity scores to see which image attained the highest score, which is the system's best guess at the identity of the individual

Open-Set Identification

Source: http://www.biometrics.gov/

Identification

Every input image has a corresponding match in the database

Biometric template of an individual is presented to the biometric system

Face recognition system:

- Compares the input image to each image in the database
- Let us assume that the similarity score for each comparison is 0.6, 0.86, 0.9, and 0.4, respectively. In this example, the correct match has the top similarity score.
- If we run the same trial for all subjects in the database, we will know how often the system will return a correct result with the top match, which is termed the identification rate at rank 1

Closed-Set Identification

Source: http://www.biometrics.gov/

Failure to Acquire

Rate at which a biometric system fails to capture and/or extract information from an observation

Numerous issues can cause a Failure to Acquire:

- Device/software malfunction
- Environmental concerns
- Human anomalies (e.g., amputees not able to use hand geometry system, bricklayers with worn fingerprints, etc.)

Biometric challenging issues will produce lower performance measures

Others only show performance (usually referred to as False Match Rates and False Non-Match Rates) on properly acquired signatures and show the Failure to Acquire rate separately

Source: http://www.biometrics.gov/

Biometric Modalities

- Dynamic Signature
- Facial Recognition
- Fingerprint
- Hand Geometry
- Iris
- Palm Print
- Speaker Recognition
- Vascular

Source: http://www.biometrics.gov/

Dynamic Signature

Source: http://www.biometrics.gov/

Dynamic Signature History

1965…first signature recognition system developed

1970s…research continues on the use of static or geometric characteristics (what the signature looks like) rather than dynamic characteristics (how the signature was made)

1970s…interest surges in dynamic characteristics with the availability of better acquisition systems accomplished through the use of touch sensitive technologies

1977…patent was awarded for a "personal identification apparatus" that was able to acquire dynamic pressure information

Source: http://www.biometrics.gov/

Dynamic Signature Technology

- Uses the anatomic and behavioral characteristics that an individual exhibits when signing his or her name (or other phrase)
- It is not a graphic image of the signature (common in locations where merchants are capturing signatures for transaction authorizations)
- Dynamically captured data:
  - Direction
  - Stroke
  - Pressure
  - Shape
- Individual's signature can enable handwriting to be a reliable indicator of an individual's identity

Source: http://www.biometrics.gov/

Face Recognition

Source: http://www.biometrics.gov/

Face Recognition History

1960s…the first semi-automated system developed:

Administrator located features (such as eyes, ears, nose, and mouth) on the photographs before it calculated distances and ratios to a common reference point

1970s…Goldstein, Harmon, and Lesk used 21 specific subjective markers such as hair color and lip thickness to automate the recognition

1988…Kirby and Sirovich applied principle component analysis

1991…Turk and Pentland discovered use of eigenfaces techniques

1993-1997…FacE REcognition Technology (FERET) Evaluation, sponsored by the Defense Advanced Research Products Agency

2000, 2002 and 2006…Face Recognition Vendor Tests (FRVT)

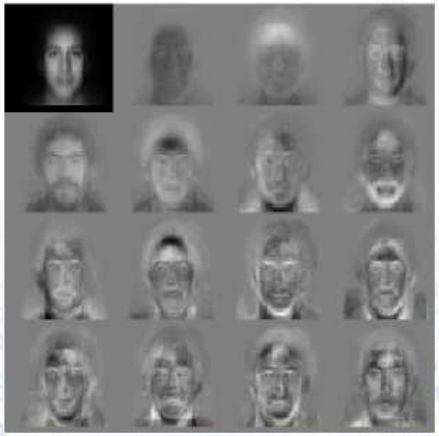
2001…NFL Super Bowl trial captured surveillance images and compared them to a database of digital mugshots

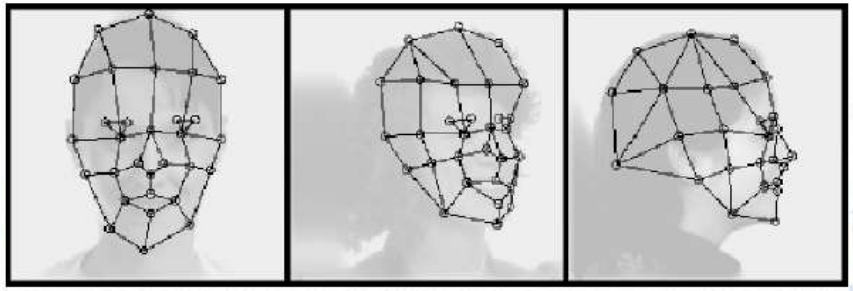
2006…***Face Recognition Grand Challenge***

Source: http://www.biometrics.gov/

Face Recognition Technologies

- **Geometric** (feature based)
- **Photometric** (view based)
- **Algorithms:**
  - Principal Components Analysis (PCA)
  - Linear Discriminant Analysis (LDA)
  - Elastic Bunch Graph Matching (EBGM)





Source: http://www.biometrics.gov/

Principal Components Analysis (PCA)

Eigenfaces (orthogonal [uncorrelated] components)

Pioneered by Kirby and Sirivich in 1988

Technique:

- Probe and gallery images are same size and normalized to line up subject's eyes and mouth in images
- PCA reduces dimension of data by compression basics and reveals the most effective low dimensional structure of facial patterns
- Dimension reduction removes useless information and precisely decomposes face structure into eigenfaces
- Each face image may be represented as a weighted sum (feature vector) of the eigenfaces, stored in a 1D array
- A probe image is compared against a gallery image by measuring the distance between their respective feature vectors

Technique reduces data needed to identify individual to 1/1000th of the data presented

PCA approach typically requires the full frontal face to be presented each time; otherwise the image results in poor performance

Source: http://www.biometrics.gov/

Linear Discriminant Analysis

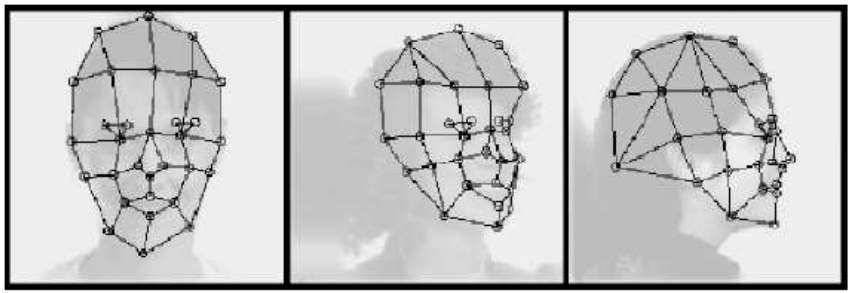
- Statistical approach for classifying samples of unknown classes based on training samples with known classes
- Aims to maximize between-class (i.e., across users) variance and minimize within-class (i.e., within user) variance.

Source: http://www.biometrics.gov/

Elastic Bunch Graph Matching

Real face images have many nonlinear characteristics:

- Variations in illumination (outdoor lighting vs. indoor fluorescents)
- Pose (standing straight vs. leaning over)
- Expression (smile vs. frown)



Technique:

- Gabor wavelet transform creates a dynamic link architecture that projects the face onto an elastic grid.
- Gabor jet is a node on the elastic grid, notated by circles on the image, which describes the image behavior around a given pixel. It is the result of a convolution of the image with a Gabor filter, which is used to detect shapes and to extract features using image processing
- Recognition is based on the similarity of the Gabor filter response at each Gabor node
- This biologically-based method using Gabor filters is a process executed in the visual cortex of higher mammals
- This method requires accurate landmark localization, which can sometimes be achieved by combining PCA and LDA methods

Source: http://www.biometrics.gov/

Fingerprinting

Source: http://www.biometrics.gov/

Fingerprinting History

Late 19th century… Sir Francis Galton defined points/characteristics to identify fingerprints.

1960s… Fingerprint identification began transition to automation

1969… Federal Bureau of Investigation (FBI) pushed to automate fingerprint identification process

1975… FBI funded development of fingerprint scanners for automated classifiers and minutiae extraction technology

1970-1980s… NIST led developments in automatic methods of digitizing inked fingerprints and the effects of image compression on image quality, classification, extraction of minutiae, and matching.

1980s… M40 algorithm, FBI's first operational matching algorithm

1981… five Automated Fingerprint Identification Systems (AFIS) deployed

1994… Integrated Automated Fingerprint Identification System (IAFIS):

1999… Lockheed Martin selected to build the AFIS segment of the FBI's IAFIS project and the major IAFIS components were operational by 1999

2003… Fingerprint Vendor Technology Evaluation (FpVTE) initiated to evaluate the accuracy of fingerprint recognition systems.

Source: http://www.biometrics.gov/

Fingerprinting Technology

Appears as a series of dark lines and white space:

- Dark lines represent the high, peaking portion of friction ridge skin
- White space is the valleys between these ridges and is the low, shallow portion of the friction ridge skin
- Based on the minutiae, or the location and direction of the ridge endings and bifurcations (splits) along a ridge path.

Information collected from a fingerprint's friction ridge impression:

- Flow of the friction ridges (Level 1 Detail)
- Presence or absence of features along the individual friction ridge paths and their sequence (Level 2 Detail)
- Intricate detail of a single ridge (Level 3 Detail)
- Recognition is usually based on Levels 1 and 2 of detail or just on Level 3

AFIS technology exploits some of these fingerprint features:

- Interprets flow of the overall ridges to assign a fingerprint classification
- Extracts minutiae detail – a subset of the total amount of information available yet enough information to effectively search a large repository of fingerprints

Source: http://www.biometrics.gov/

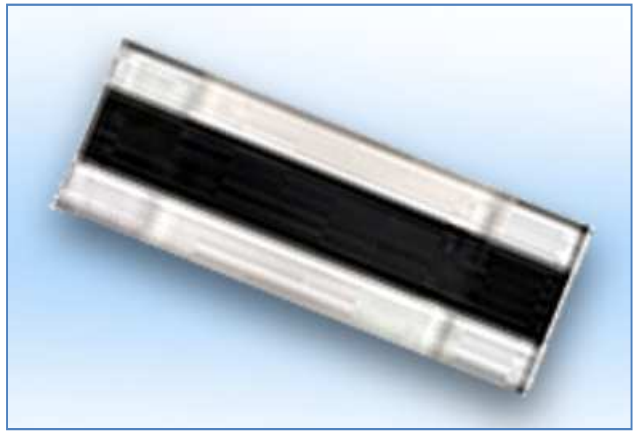
Fingerprint Sensor

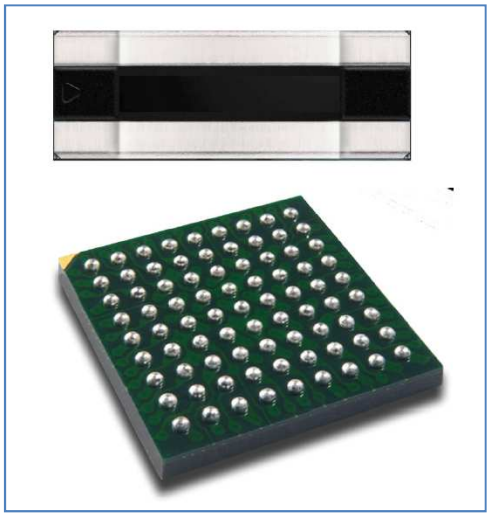
- Optical sensors take an image of the fingerprint, and are the most common sensor today
- Capacitive sensors determine each pixel value based on the capacitance measured, made possible because an area of air (valley) has significantly less capacitance than an area of finger (friction ridge skin)
- Ultrasound employs high frequency sound waves
- Thermal requires a swipe of a finger across a surface to measure the difference in temperature over time to create a digital image

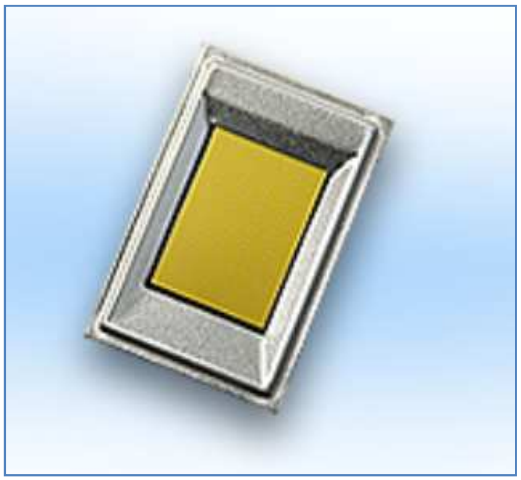
Source: http://www.biometrics.gov/

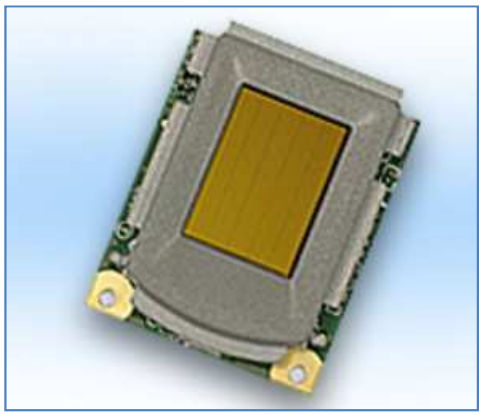
Sensors USB

Chipset





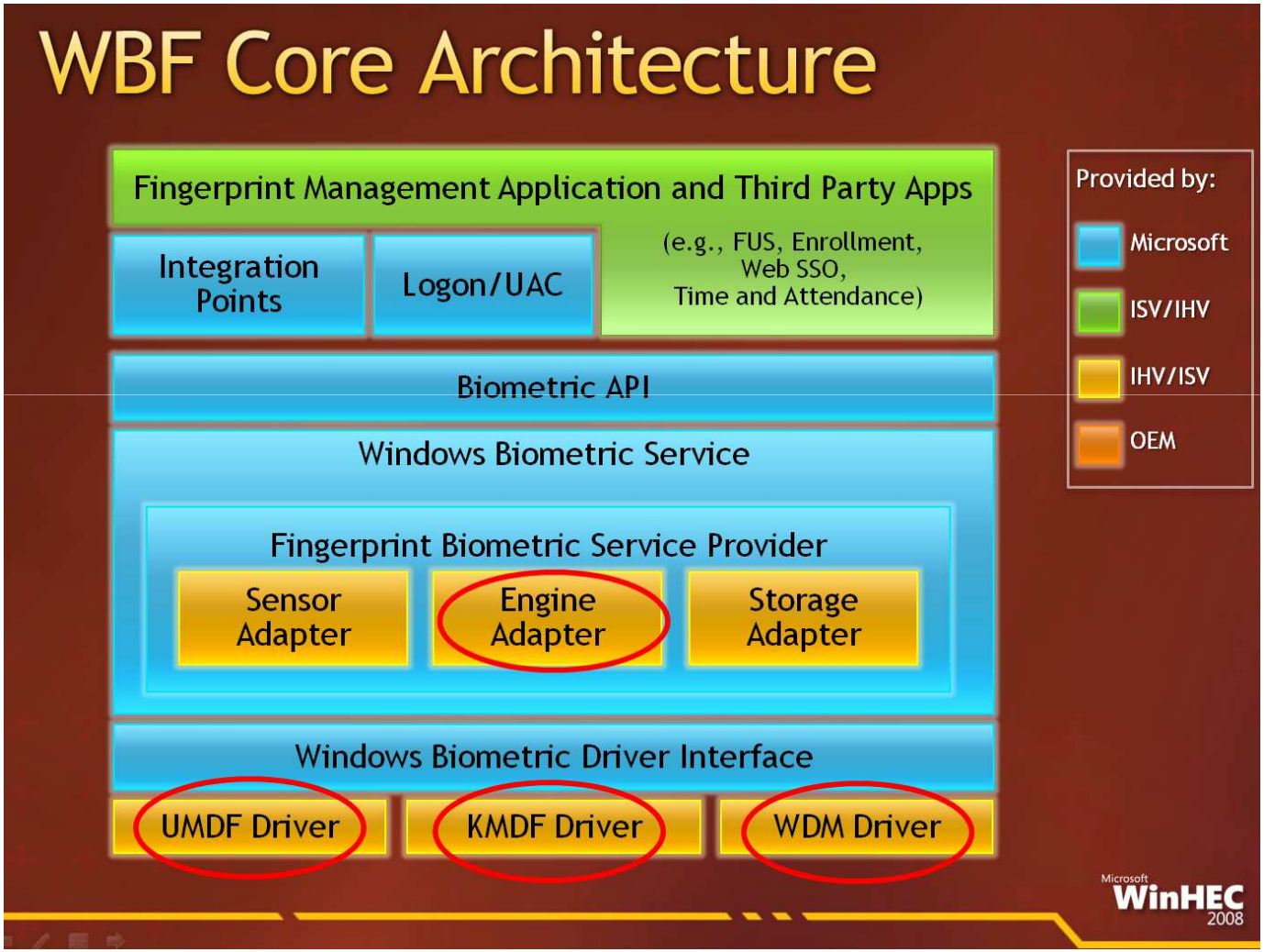




PIV-FIPS 201 Sensors

Tablet approach

Windows Biometric Framework



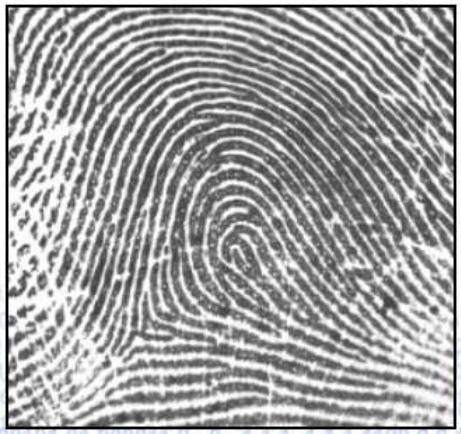
Fingerprint Software

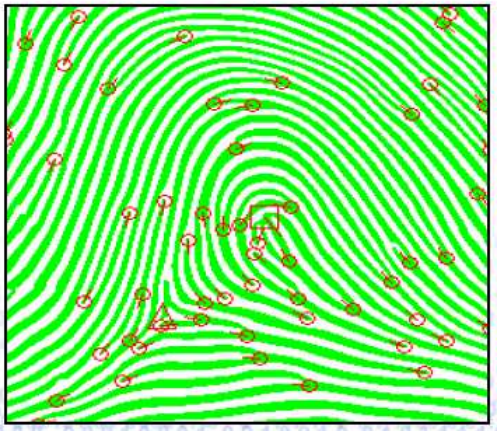
Minutiae-based matching:

- Relies on locations and direction of minutiae points
- Most widely used matching technique

Pattern matching:

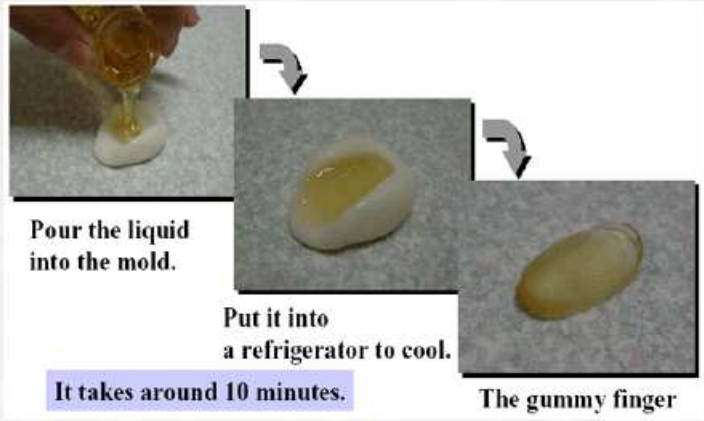
- Compares two images to judge their similarity
- Used to detect duplicates

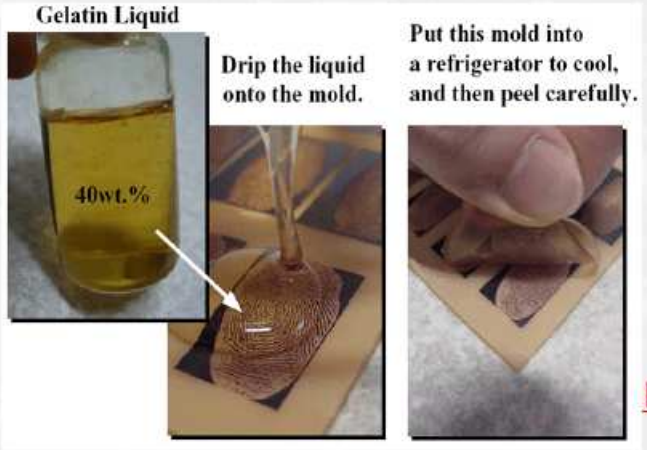




Source: http://www.biometrics.gov/

Matsumoto's « Gummy Fingers »

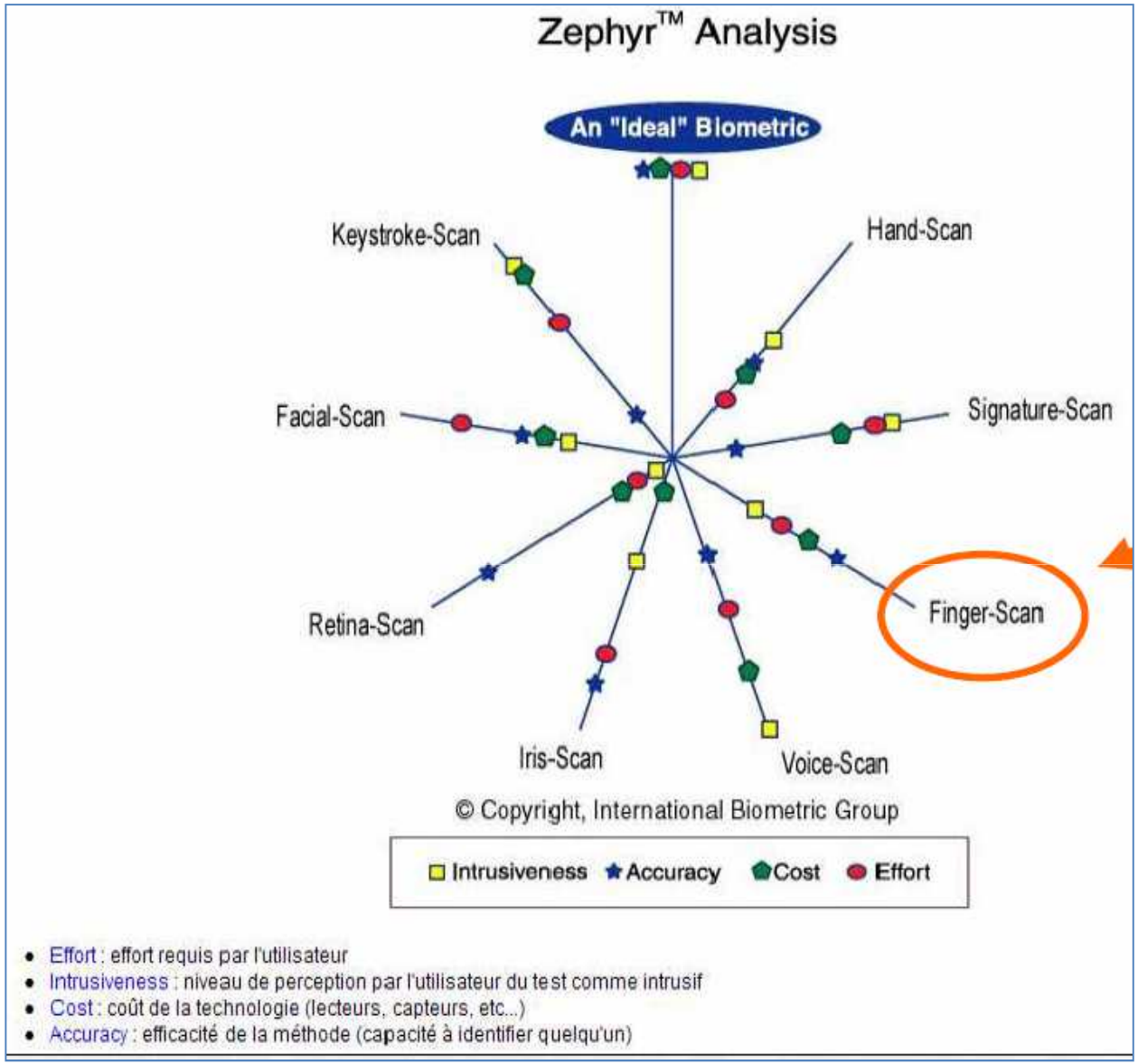




Etude Yokohama University

http://crypto.csail.mit.edu/classes/6.857/papers/gummy-slides.pdf

e-xpert solutions Le bon sens et l'expérience


Zephyr™ Analysis
An "Ideal" Biometric
Keystroke-Scan
Hand-Scan
Facial-Scan
Signature-Scan
Retina-Scan
Finger-Scan
Iris-Scan
Voice-Scan
© Copyright, International Biometric Group
Intrusiveness
Accuracy
Cost
Effort
Effort : effort requis par l'utilisateur
Intrusiveness : niveau de perception par l'utilisateur du test comme intrusif
Cost : coût de la technologie (lecteurs, capteurs, etc...)
Accuracy : efficacité de la méthode (capacité à identifier quelqu'un)

Hand Geometry

Source: http://www.biometrics.gov/

Hand Geometry History

1980s…Hand geometry introduced

1985…David Sidlauskas developed and patented the hand geometry

1986…First commercial hand geometry recognition systems

1991…Performance evaluation of biometric identification devices evaluated the relative performance of multiple biometric devices, including hand geometry

1996…Olympic Games implemented hand geometry systems to control and protect physical access to the Olympic Village

1996…Evaluation of the INSPASS Hand Geometry Data determined the effect of a threshold on system operation, established false accept and false reject rates as a function of the threshold, and presented an estimate of the Receiver Operating Characteristics (ROC) curve for the INSPASS system

1990s-present…Companies implement hand geometry systems in parallel with time clocks for time and attendance purposes.

2004…Walt Disney World began using "finger" geometry to expedite and facilitate entrance to the park and to identify guests as season ticket holders to prevent ticket fraud

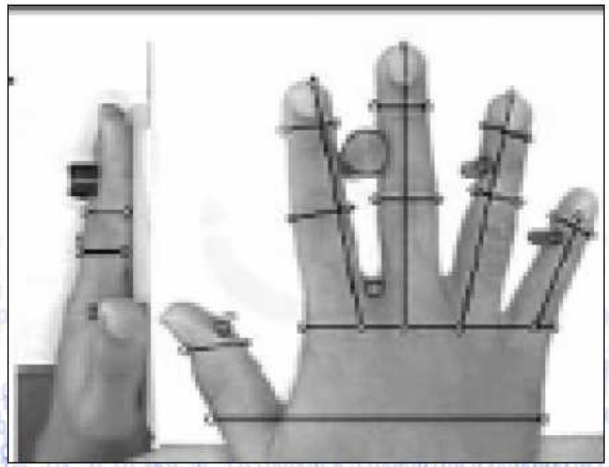
Source: http://www.biometrics.gov/

Hand Geometry History

Use measurement and recording length, width, thickness, and surface area of an individual's hand while guided on a plate

Use camera to capture hand silhouette

- Hand placed on the plate, palm down, and guided by five pegs that sense when the hand is in place
- Data capture by a Charge-Coupled Device (CCD) camera of the top view of the hand, top surface of the hand and a side image (using an angled mirror)
- 31,000 points are analyzed and 90 measurements are taken
- Information is stored in nine bytes of data



Source: http://www.biometrics.gov/

Hand Geometry Technology

Enrollment process:

- Requires the capture of three sequential images of the hand
- Creates a template of the user's characteristics

Submission process:

- System recalls the template associated with that identity
- Claimant places his/her hand on the plate
- System captures an image and creates a verification template to compare to the template developed upon enrollment
- Similarity score is produced
- Claimant accepted or rejected based on system threshold

Source: http://www.biometrics.gov/

Iris Recognition

Source: http://www.biometrics.gov/

Iris Recognition History

1936…Ophthalmologist Frank Burch proposed using iris patterns as a method to recognize an individual

1985…Drs. Leonard Flom and Aran Safir, ophthalmologists, proposed the concept that no two irides are alike

1987…Drs. Leonard Flom and Aran Safir awarded a patent for the iris identification concept

1993…Defense Nuclear Agency began work to test and deliver a prototype unit

1994…Dr. Daugman awarded a patent for his automated iris recognition algorithms

1995…Prototype completed due to the combined efforts of Drs. Flom, Safir, and Daugman

1995…First commercial products

2005…Patent covering the basic concept of iris recognition expired, providing marketing opportunities for other companies that have developed their own algorithms for iris recognition

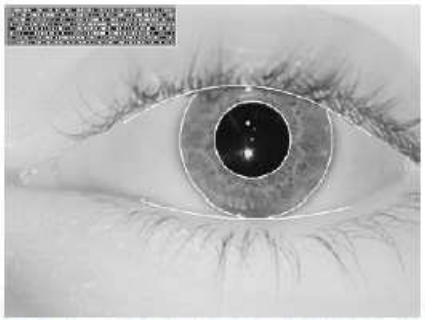
2011…Expiration of patent on the IrisCodes® implementation of iris recognition developed by Dr. Daugman

Source: http://www.biometrics.gov/

Iris Recognition Technology

Iris imaging requires use of a high-quality digital camera:

- Commercial iris cameras use infrared light to illuminate the iris without causing harm or discomfort to the subject
- 2D Gabor wavelet filters and maps the segments of the iris into phasors (vectors)
- Phasors include information on the orientation and spatial frequency ("what" of the image) and the position of these areas ("where")
- This information is used to map the IrisCodes®



Source: http://www.biometrics.gov/

Iris Recognition Technology

Phase is not affected by contrast, camera gain, or illumination levels:

- Uses 256 bytes of data using a polar coordinate system
- Control bytes exclude eyelashes, reflection(s), and other unwanted data

Recognition performed comparing two IrisCodes®

- Amount of difference between two IrisCodes® — Hamming Distance (HD) — is used as a test of statistical independence between the two IrisCodes®. If the HD indicates that less than one-third of the bytes in the IrisCodes® are different, the IrisCode® fails the test of statistical significance, indicating that the IrisCodes® are from the same iris.

Therefore, the key concept to iris recognition is failure of the test of statistical independence.

Source: http://www.biometrics.gov/

Palm Print

Source: http://www.biometrics.gov/

Palm Print History

1858…Sir William Herschel, working for the Civil Service of India, recorded a handprint on the back of a contract for each worker to distinguish employees from others who might claim to be employees when payday arrived

1994…First known AFIS system to support palm prints is believed to have been built by a Hungarian company.

Late 1997…US company bought Hungarian palm system

2000s…Australia houses the largest repository of palm prints in the world. The new Australian National Automated Fingerprint Identification System (NAFIS) includes 4.8 million palm prints

April 2002…a Staff Paper on palm print technology and IAFIS palm print capabilities was submitted to the Identification Services (IS) Subcommittee, CJIS Advisory Policy Board (APB)

2004…Connecticut, Rhode Island and California established statewide palm print databases that allowed law enforcement agencies in each state to submit unidentified latent palm prints to be searched against each other's database of known offenders

Source: http://www.biometrics.gov/

Palm Print Technology

Palm print:

- Series of dark lines represents the high, peaking portion of the friction ridged skin
- White space represents the valley between these ridges
- Interprets the flow of the overall ridges to assign a classification and then extract the minutiae detail — a subset of the total amount of information available, yet enough information to effectively search a large repository of palm prints

Sensor types:

- Capacitive determines each pixel value based on the capacitance measured
- Optical uses prisms to detect the change in light reflectance related to the palm
- Ultrasound employs high frequency sound
- Thermal requires a swipe of a palm across a surface to measure the difference in temperature

Source: http://www.biometrics.gov/

Palm Print Technology

Software: scan the entire palm or segment it into smaller areas

- Palm systems partition their repositories based upon the location of a friction ridge area
- Searching only this region of a palm repository rather than the entire database maximizes the reliability of a latent palm search

Palm matching techniques

- Minutiae-based matching relies on the minutiae points described above, specifically the location, direction, and orientation of each point
- Correlation-based matching involves simply lining up the palm images and subtracting them to determine if the ridges in the two palm images correspond
- Ridge-based matching uses ridge pattern landmark features such as sweat pores, spatial attributes, and geometric characteristics of the ridges, and/or local texture analysis, all of which are alternates to minutiae characteristic extraction matching

Source: http://www.biometrics.gov/

Speaker Verification

Speaker Verification History

1960…Gunnar Fant, a Swedish professor, published a model describing the physiological components of acoustic speech production, based on the analysis of x-rays of individuals making specified phonic sounds

1970…Dr. Joseph Perkell used motion x-rays and included the tongue and jaw to expand upon the Fant model

1976…Texas Instruments built a prototype system that was tested by the U.S. Air Force and The MITRE Corporation

Mid-1980s…the National Institute of Standards and Technology (NIST) developed the NIST Speech Group to study and promote the use of speech processing techniques

1996...National Security Agency funded and the NIST Speech Group has hosted yearly evaluations, the NIST Speaker Recognition Evaluation Workshop, to foster the continued advancement of the speaker recognition community

Source: http://www.biometrics.gov/

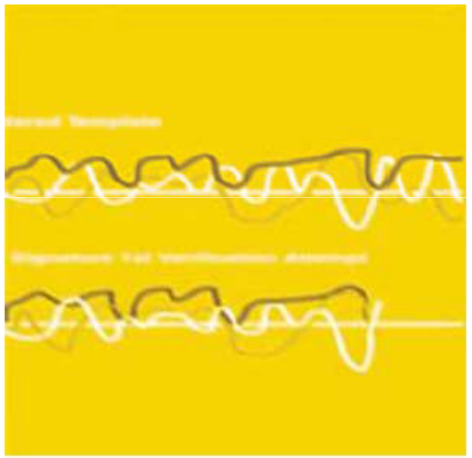
Speaker Verification Technology

Physiological component of voice recognition related to physical shape of an individual's vocal tract

Motion of the mouth and pronunciations are the behavioral components of this biometric

There are two forms of speaker recognition:

- **Text dependent** *(constrained mode):*
  - Individual presents either a fixed (password) or is prompted for a phrase *("Please say the numbers '33-54-63'")* that is programmed into the system and can improve performance especially with cooperative users
- **Text independent** *(unconstrained mode):*
  - No advance knowledge of the presenter's phrasing and is much more flexible in situations where the individual submitting the sample may be unaware of the collection or unwilling to cooperate, which presents a more difficult challenge



Source: http://www.biometrics.gov/

Speaker Verification Technology

Text dependent:

- Capture word or phrase by microphone
- Convert voice sample from analog to digital to extract voice features, and create a model
- Use Hidden Markov Models (HMMs), random based models that provide a statistical representation of the sounds produced by the individual

Text independent:

- Uses Gaussian Mixture Model, a state-mapping model closely related to HMM
- Uses the voice to create vector "states" representing sound forms characteristic of the person's physiology and behavior

Enrollment

Source: http://www.biometrics.gov/

Speaker Verification Technology

- Same quality/duration/loudness/pitch features are extracted from the submitted sample and compared to the model of the claimed or hypothesized identity and to models from other speakers
- Other-speaker models contain the "states" of a variety of individuals, not including that of the claimed or hypothesized identity
- Input voice sample and enrolled models are compared to produce a likelihood ratio
- If the voice input belongs to the identity claimed or hypothesized, the score will reflect the sample to be more similar to the claimed identity's model than to the "anti-speaker" model

Recognition

Source: http://www.biometrics.gov/

Vascular Pattern

Vascular Pattern History

1992…Dr. K. Shimizu published paper on optical trans-body imaging and potential optical CT scanning applications

1996…K. Yamamoto, in conjunction with K. Shimizu, presented another paper in which the two discussed research they had undertaken since the earlier paper

2000…First research paper about the use of vascular patterns for biometric recognition published

2000…First commercially available device using subcutaneous blood vessel pattern in the back of the hands

Source: http://www.biometrics.gov/

Vascular Pattern Technology

Vascular hand pattern collection:

- Near-infrared rays from a bank of light emitting diodes (LEDs) penetrate the skin of the back or palm of the hand
- Reflected near-infrared rays produce an image on the sensor caused by absorbance of blood vessels and other tissues
- Image is digitized and image processing techniques produce extracted vascular pattern
- Various feature make up template:
  - Vessel branching points
  - Thickness
  - Branching angles

Source: http://www.biometrics.gov/

Vascular Pattern Technology

Vascular finger pattern collection:

- Near-infrared rays generated from a bank of LEDs penetrate the finger or hand and are absorbed by the hemoglobin in the blood
- Areas in which the rays are absorbed (i.e., veins) appear as dark areas similar to a shadow in an image taken by a charge-coupled device (CCD) camera
- Image processing can then construct a vein pattern from the captured image
- Pattern is digitized and compressed so that it can be registered as a template

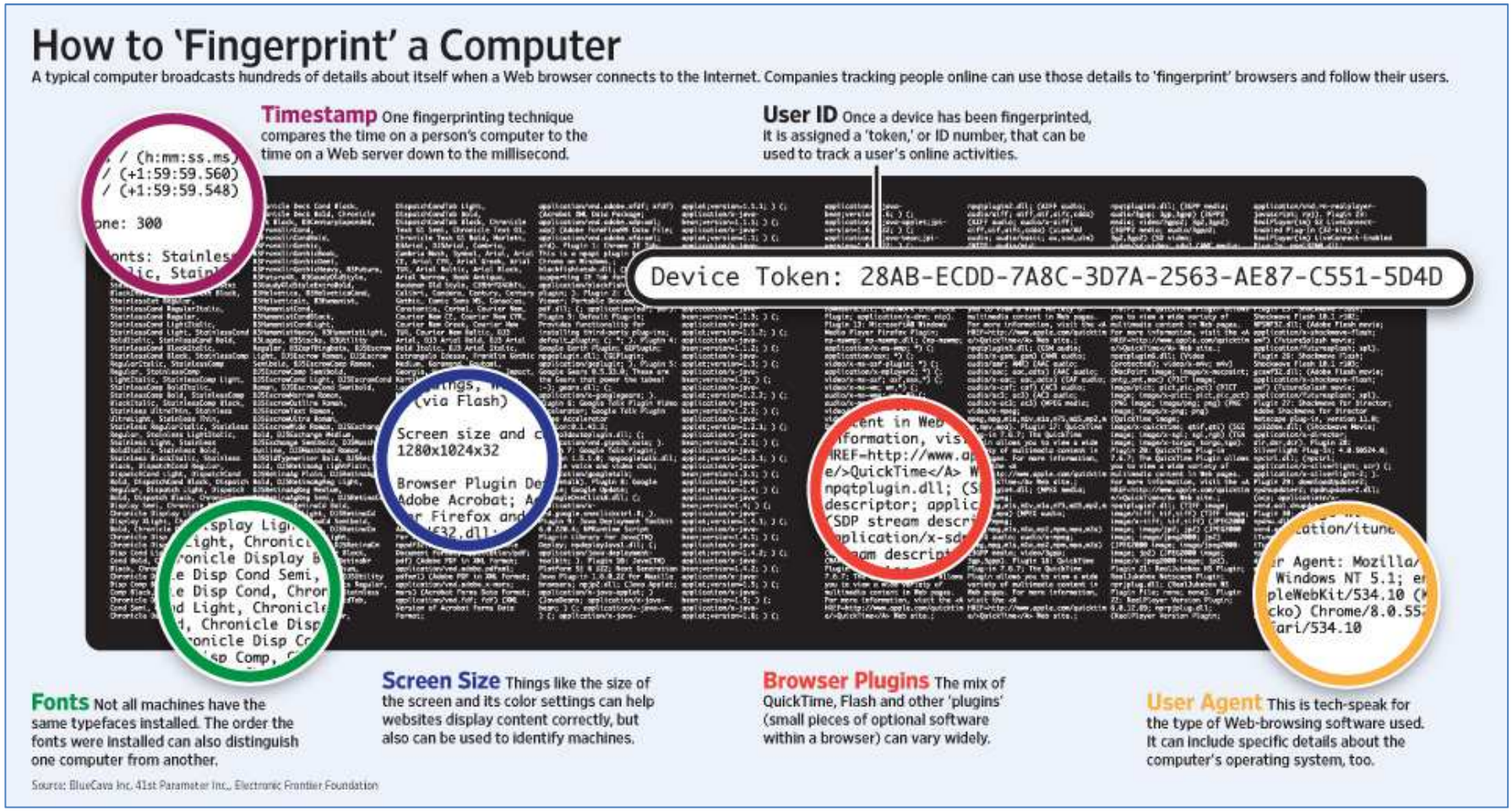
Source: http://www.biometrics.gov/

Vascular Pattern Technology

Device fingerprint - DNA

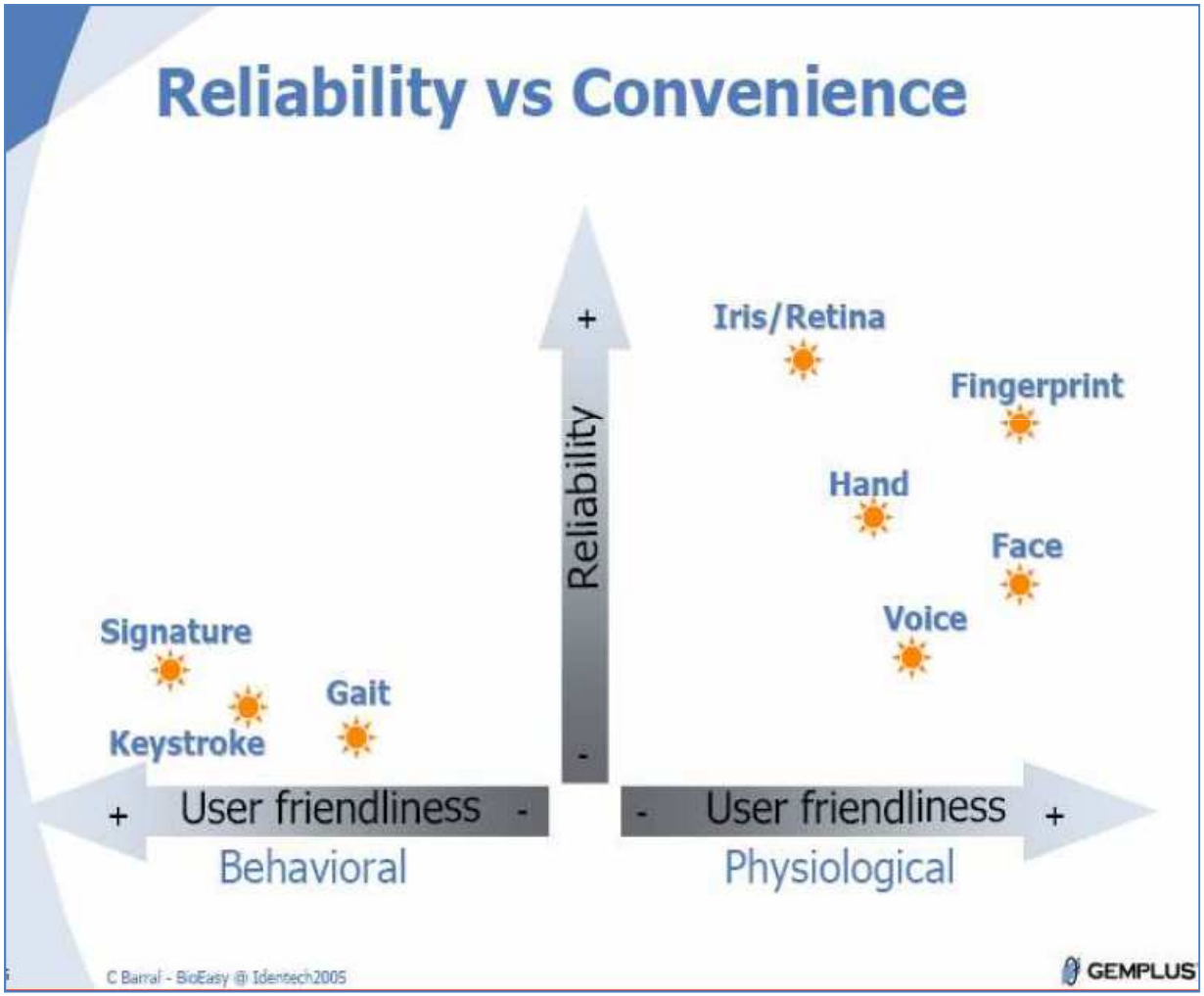
- A device fingerprint or machine fingerprint or browser fingerprint is information collected about a remote computing device for the purpose of identification

Fingerprint a Computer



Source = The Wall Street Journa

Biometrics Technology



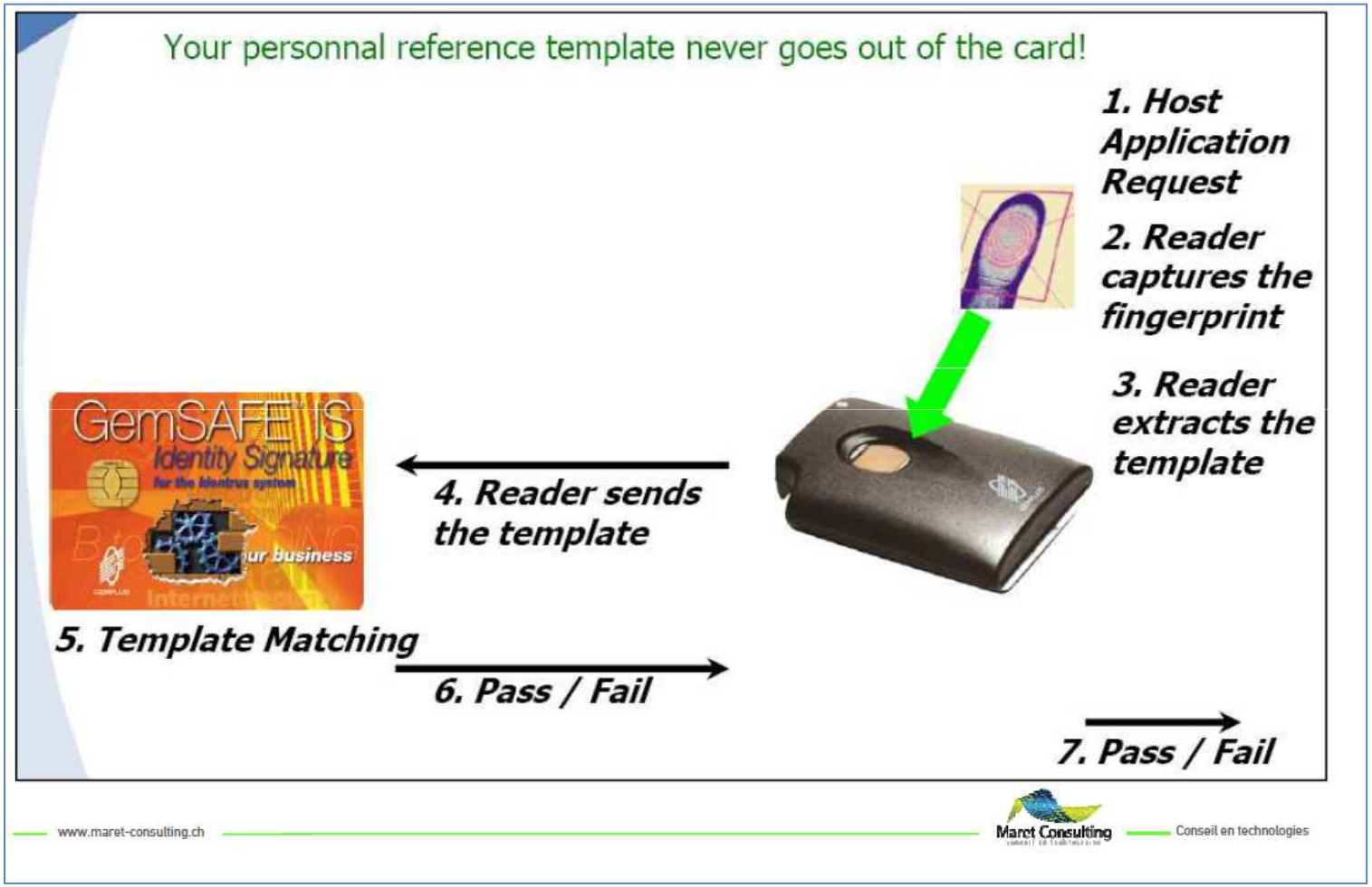
Match-on-Card

Strong Authentication with Biometry (Match on Card technology)

- A reader
  - Biometry
  - SmartCard

- A card with chip
  - Technology MOC
  - Crypto Processor
    - PC/SC
    - PKCS#11
    - Digital certificate X509

MOC


Your personnal reference template never goes out of the card!
1. Host Application Request
2. Reader captures the fingerprint
3. Reader extracts the template
4. Reader sends the template
GemSAFE IS
Identity Signature
5. Template Matching
6. Pass / Fail
7. Pass / Fail
www.maret-consulting.ch
Maret Consulting
Conseil en technologies

MOC – Athena & Precise Biometrics

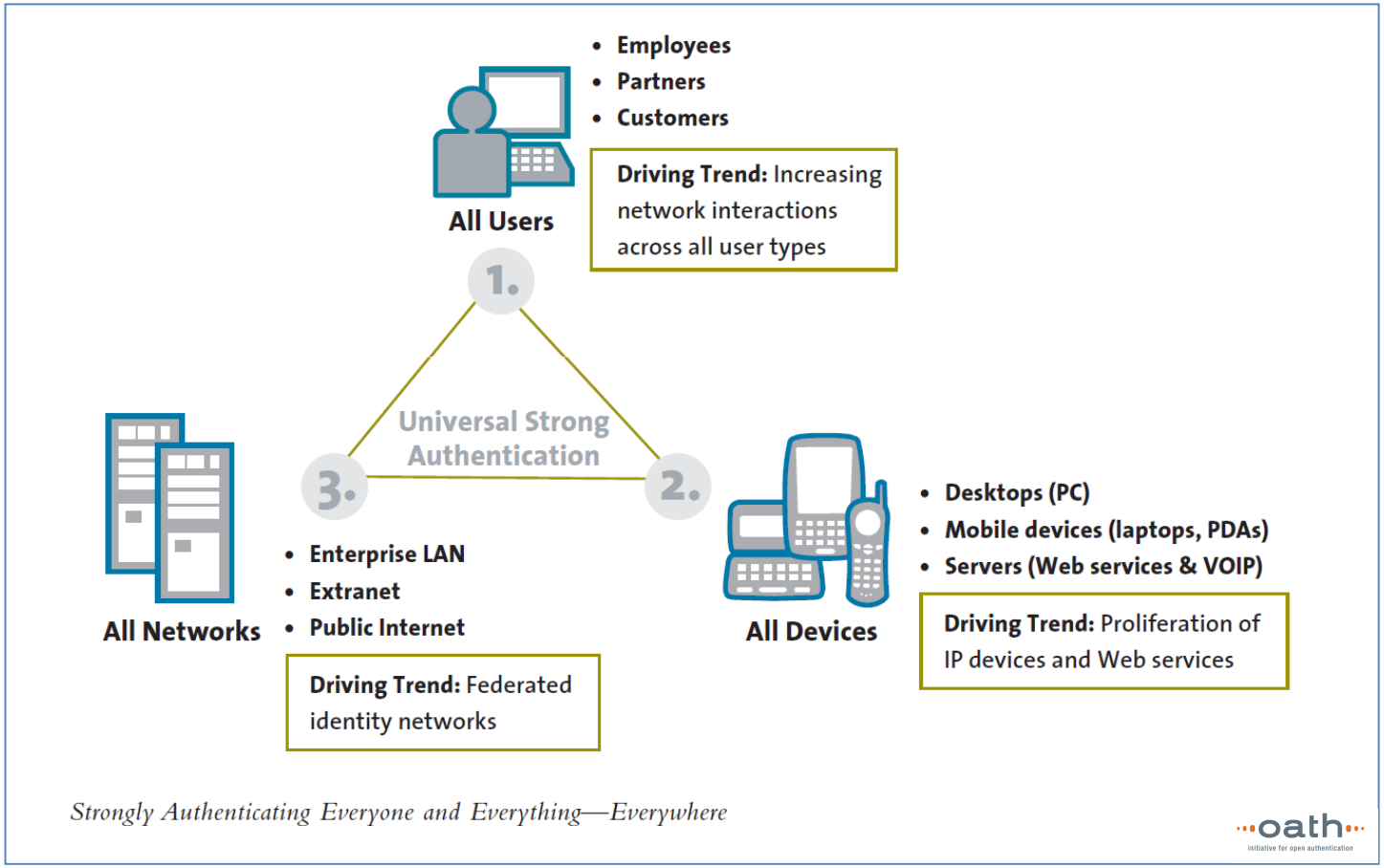
	OTP	PKI (HW)	Biometry
Strong authentication			
Encryption			
Digital signature			
Non repudiation			
Strong link with the user			

initiative for open authentication

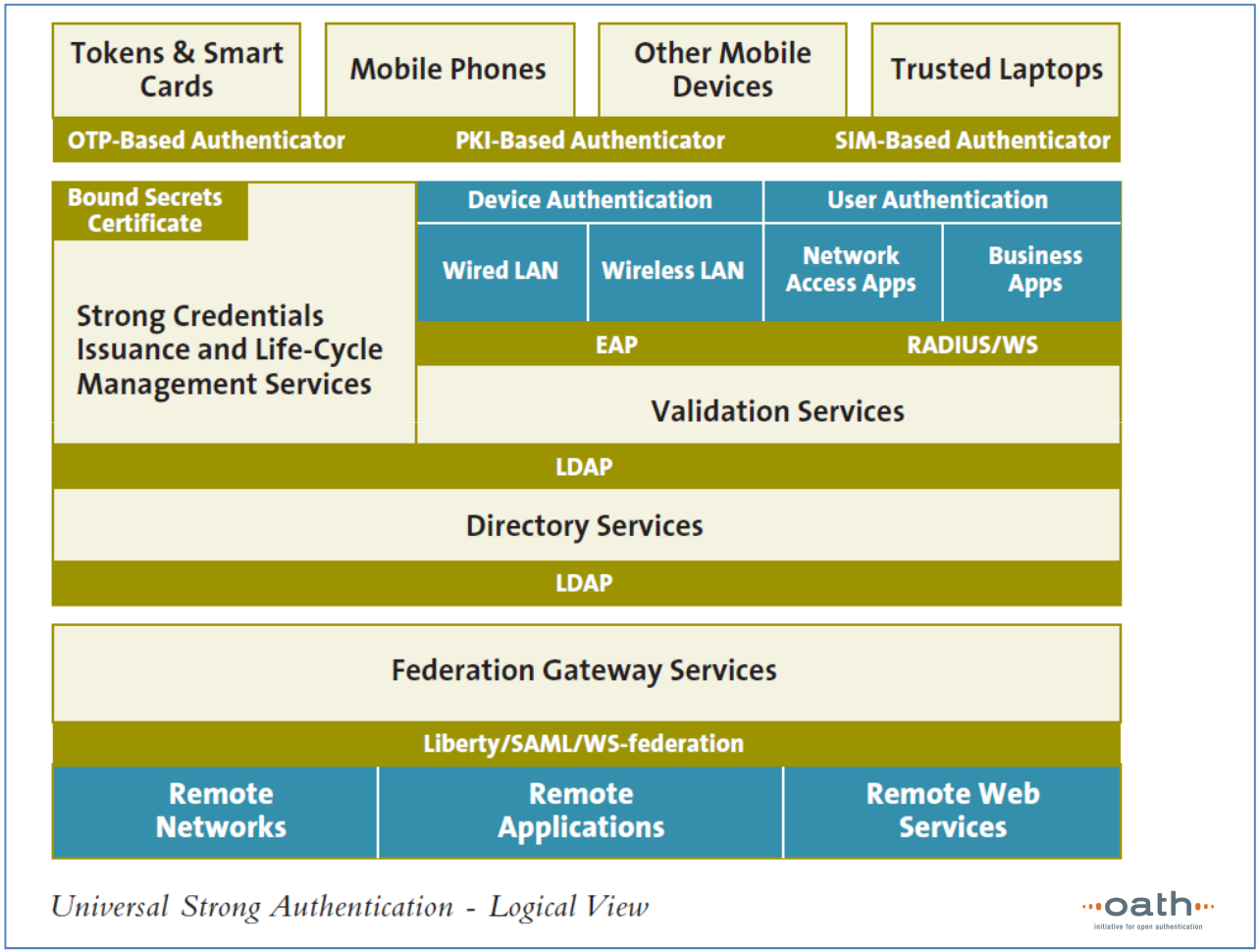
OATH approach

Open Authentication

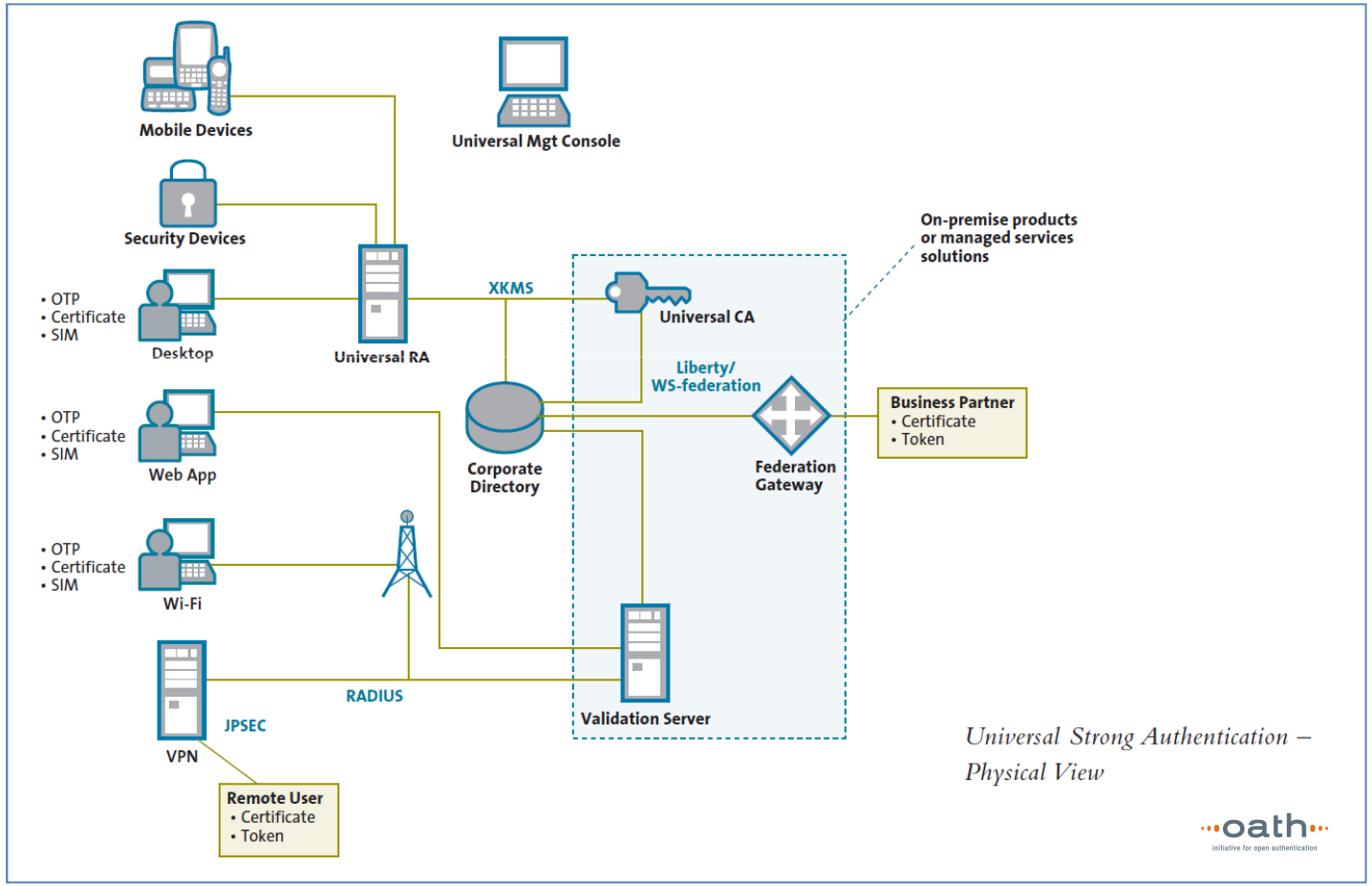
OATH Approach



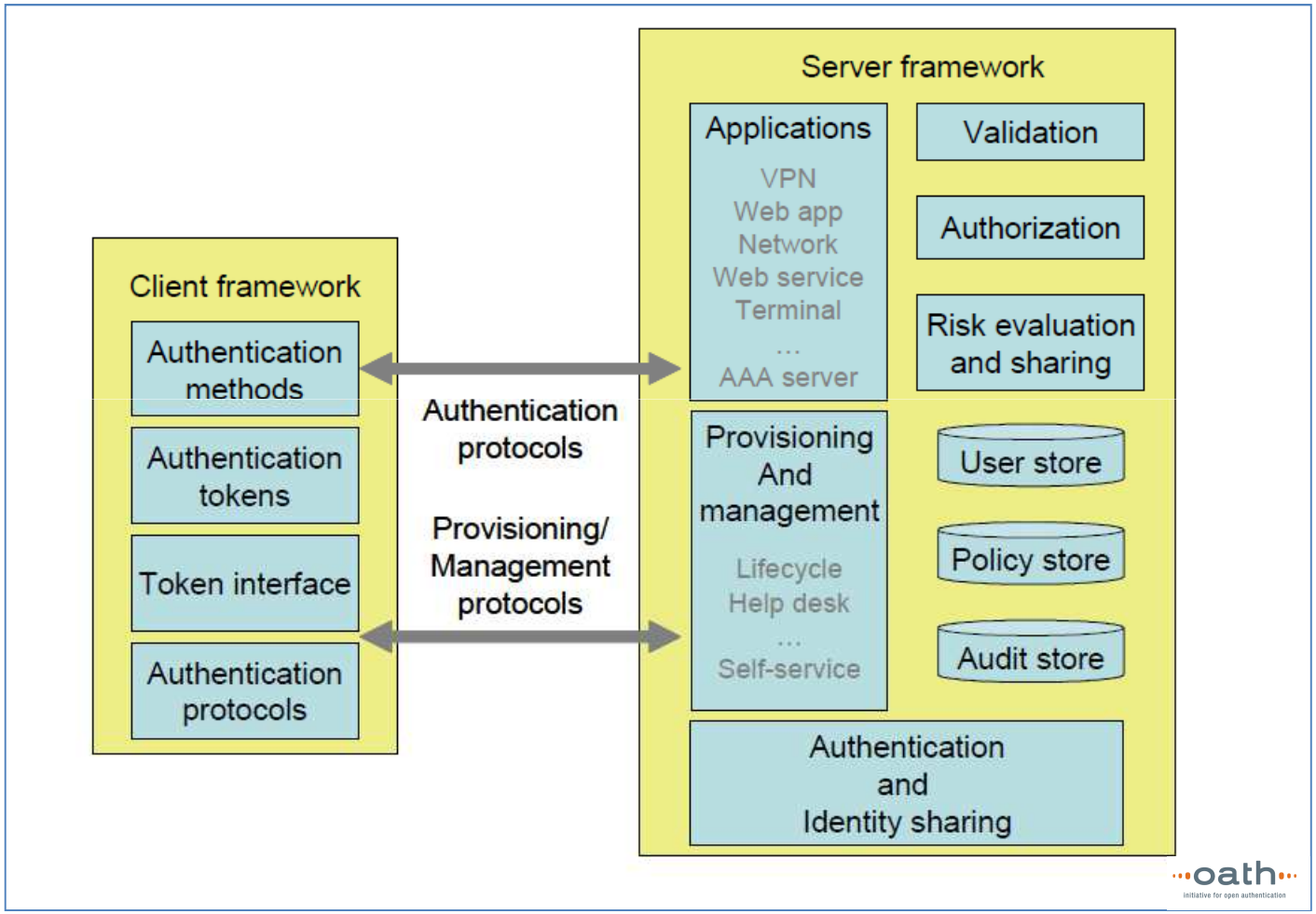
OATH Logical view



OATH Physical view



OATH Authentication Framework



OATH Client framework

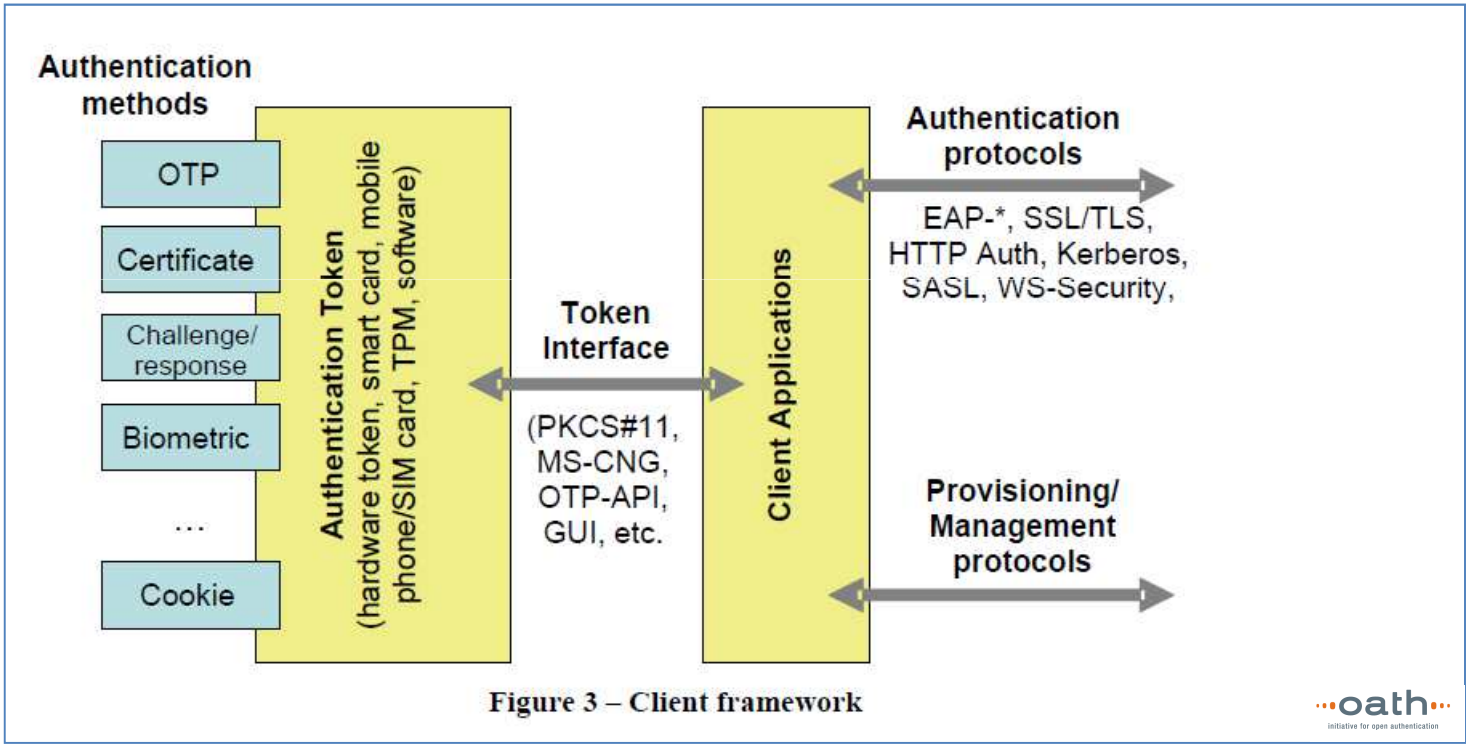


Figure 3 – Client framework

OATH AuthN methods 1/2

Existing Standards and Technologies

Password – The password authentication method is the oldest and still most commonly used method for user authentication. However, password authentication is no longer considered adequately secure in many applications, because users can share their passwords and because replay attacks have become common. Password authentication, on its own, is not recommended by OATH. But, in order to provide a more complete picture of available methods, it is described here. The password authentication method is based on data matching. The user is identified by a user ID and their authentication password is matched with the corresponding data stored by the validation service for that user.

One-Time-Password – The one-time-password, or OTP, authentication method can be divided into two sub-types. Time-based methods rely on the transformation of a shared secret and a time value that is synchronized between the server and the client. Event-based methods rely on the transformation of a shared secret and an event count that is synchronized between the server and the client. Typically, the event that is counted is the pressing of a button on the token. [HOTP]

Challenge / Response – The challenge / response authentication method is usually based on a shared-secret transformation using symmetric-key hashing techniques. The server side sends the client a challenge, and the client uses this challenge and the shared secret as input to the transformation. The resulting output is the response, which is then sent to the server.

oath
initiative for open authentication

OATH AuthN methods 2/2

Transaction Signing – This method is a more advanced version of challenge / response, where the user confirms certain details of the transaction (e.g. target account number, amount and currency, for a funds transfer transaction). These details are then input into the algorithmic computation, often based on symmetric cryptography. EMV-CAP Mode 2 with TDS [CAP04], APACS prompted Data Signing or OCRA are examples of algorithms that support transaction signing. Usually the server would transmit the specific details and the user would type them into a token or an unconnected smart card reader. The token / reader would then display a response that is sent to the server for verification.

User certificate – Certificate-based authentication uses public-key encryption techniques, supported by a public-key infrastructure (PKI) for key and certificate management. Digital certificates are issued by a certification authority and they bind the user's identity to their public key. In a typical certificate-based authentication protocol, the client uses its private key to sign a challenge from the server, and the server verifies the signature using the client's certificate. [PKIX]

Biometric - Biometric authentication methods are based on a physiological characteristic of a user, such as a fingerprint, iris image or facial image. Biometric authentication represents the "what you are" component of multi-factor authentication. Biometric authentication is based on data-matching of the captured biometric characteristic to a stored template.

Device fingerprint – A Web application may examine a persistent cookie, the source IP address and the type and version of the remote user agent. This information can be used to identify suspected impersonation attacks. However, because users legitimately change or update their browsers periodically, this technique is subject to "false positives". Nevertheless, it can be used as one in a set of risk metrics to decide when step-up authentication is required.

oath
initiative for open authentication

OATH AuthN protocols 1/3

Existing Standards and Technologies

Providing a comprehensive framework for authentication services requires that we first have an understanding of protocols and mechanisms currently in use. Listed below are the most common authentication protocols with short descriptions of their use and references to more detailed descriptions.

Challenge Handshake Authentication Protocol - CHAP is an authentication protocol used to log a user on to an Internet access provider. It was widely used in early dialup services. See [RFC1334] and [RFC1994].

Extensible Authentication Protocol - EAP is used between a dialup client and a server to determine what authentication protocol will be used. EAP is also widely used for other client / server authentication services. See [RFC3748].

Generic Security Service Application Program Interface – GSSAPI provides security services to calling applications in a generic fashion, supported by a range of underlying mechanisms and technologies. It allows source-level portability of applications to different environments. The authentication method specified by GSSAPI is very generic and further defined in other RFCs that build on GSSAPI. See [RFC1508].

oath
initiative for open authentication

OATH AuthN protocols 2/3

HTTP Basic and Digest Authentication - HTTP authentication describes username / password authentication for HTTP 1.1. It is commonly used in combination with SSL to provide confidentiality for the password (Basic) or a cryptographic hash of the password (Digest) as it is sent over the channel. See [RFC2617]

Kerberos – The Kerberos network authentication protocol is used in a distributed computing environment. It is based on the principle that the user authenticates to an authentication server. The server then grants the user the right to request tickets from one or more ticket granting servers that issue authentication tickets for any application that the user has the right to access. Kerberos acts much like a single-sign-on solution between applications and trusted computers. Kerberos is, for example, used in Microsoft Windows 2000 and above. See [RFC1510].

MSCHAP v1 and v2 - Microsoft's PPP CHAP dialect (MSCHAP) extends the user authentication functionality provided on Windows networks to remote workstations. MSCHAP is derived from the PPP Challenge Handshake Authentication Protocol. See [RFC2433] and [RFC2759].

Password Authentication Protocol – PAP is a two-way handshake protocol designed for use with PPP. PAP is a plain-text password protocol used on older SLIP systems. It is not considered secure, because it passes the credentials in clear text. See [RFC1334] and [RFC1994].

Simple Authentication and Security Layer - SASL defines a method for adding authentication support to connection-based protocols. See [RFC2222].

oath
initiative for open authentication

OATH AuthN protocols 3/3

SSL/TLS - The transport layer security protocol is widely supported in standard Internet browsers and Web servers. It is based on digital certificates, and can provide mutual authentication. See [RFC2246].

WS Security – The OASIS Web Services Security specification describes enhancements to SOAP messaging that protect message integrity, confidentiality and source authentication. These mechanisms can be used with a wide variety of security models and encryption technologies. WS-Security also provides a general-purpose mechanism for associating security tokens with messages. See [WSS].

Sideband signaling - In certain types of man-in-the-middle and Trojan attacks, the adversary impersonates the user by using the user's genuine credentials on the user's own device. Sideband signaling may be used to mitigate this threat. Sideband signaling can be directly integrated into the authentication protocol. This is a particularly appealing approach in situations where wireless devices and wired devices are both participating in the authentication process, thereby providing multiple IP communications paths to the user's location. In addition to direct integration into the authentication protocols, sideband signaling may be used to select the authentication protocol to be used. Sideband signaling can further mitigate these attacks by requiring explicit out-of-band notification and / or confirmation of high-value transactions.

oath
initiative for open authentication

OATH AuthN validation framework

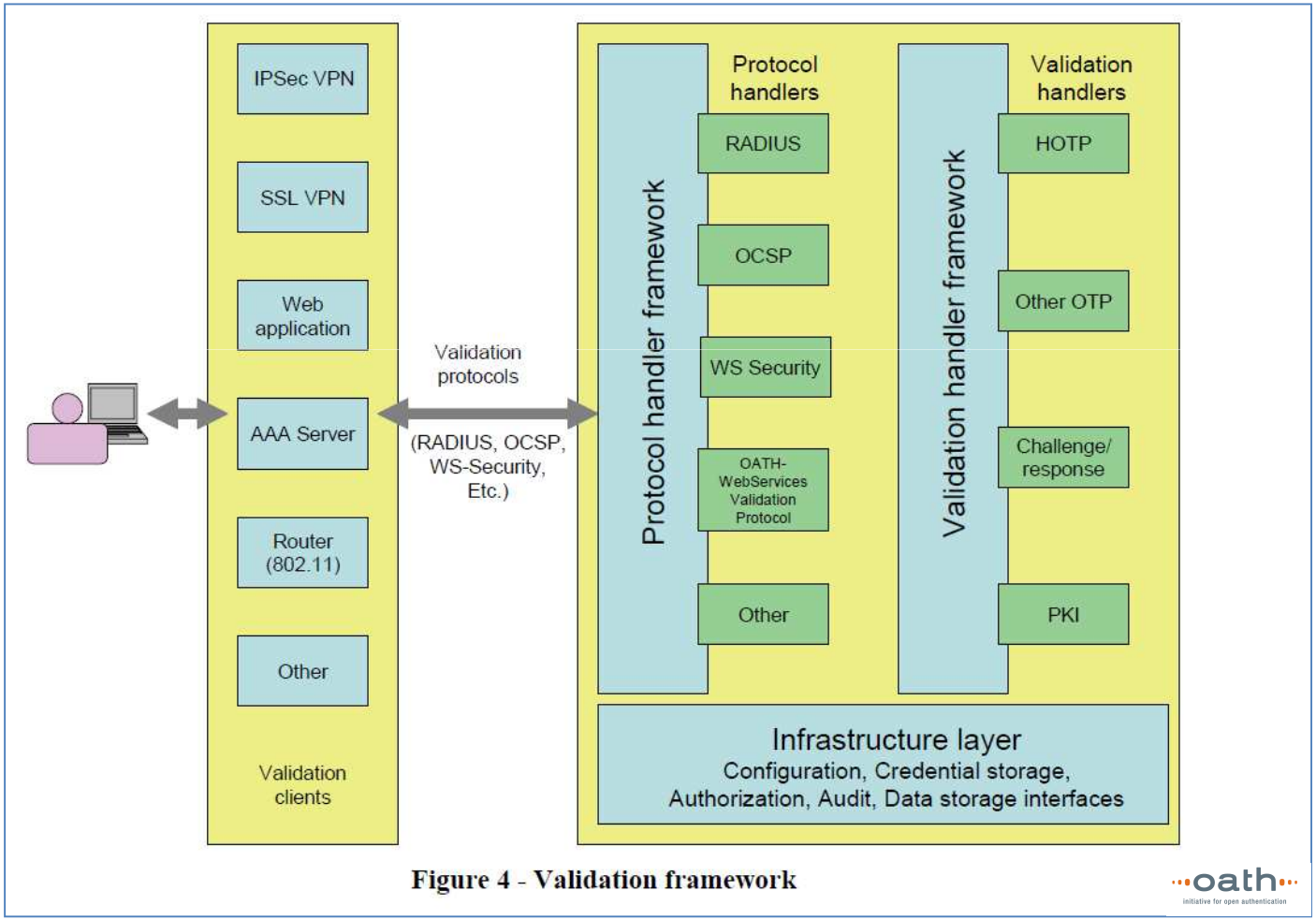


Figure 4 - Validation framework

OATH validation protocols

Existing Validation Protocols

Lightweight Directory Access Protocol - LDAP is a protocol for accessing on-line directory services. LDAP defines a relatively simple protocol for updating and searching directories over TCP/IP. Directories are used to store information about end-users, including usernames and passwords. Consequently, LDAP is often used by applications to validate the username and password of an end-user.

Online Certificate Status Protocol - OCSP is a method for determining the revocation status of an X.509 digital certificate without directly using a CRL. OCSP's request / response nature leads to OCSP servers being called OCSP responders. See [RFC2560].

Remote Authentication Dial In User Service - RADIUS is an authentication, authorization and accounting (AAA) protocol for applications such as network access, which can be used in both local and roaming situations. See [RFC2138].

Diameter - Similar to RADIUS, Diameter is an authentication, authorization and accounting (AAA) protocol for applications such as network access or IP mobility. See [RFC3539].

Terminal Access Controller Access-Control System - TACACS is a remote authentication protocol. It is commonly used in UNIX networks. It allows a remote access server to communicate with an authentication server in order to determine if the user should be allowed access to the network. See [RFC1492].

XML Key Management Specification - XKMS provides a secure method for registration and subsequent life-cycle management of public-key information. The XML Key Management Specification (XKMS) comprises two parts: the XML Key Information Service Specification (XKISS) and the XML Key Registration Service Specification (XKRSS). XKISS may be used to validate a signature and the certificate associated with that signature. See [XKMS].

oath
initiative for open authentication

OATH provisioning

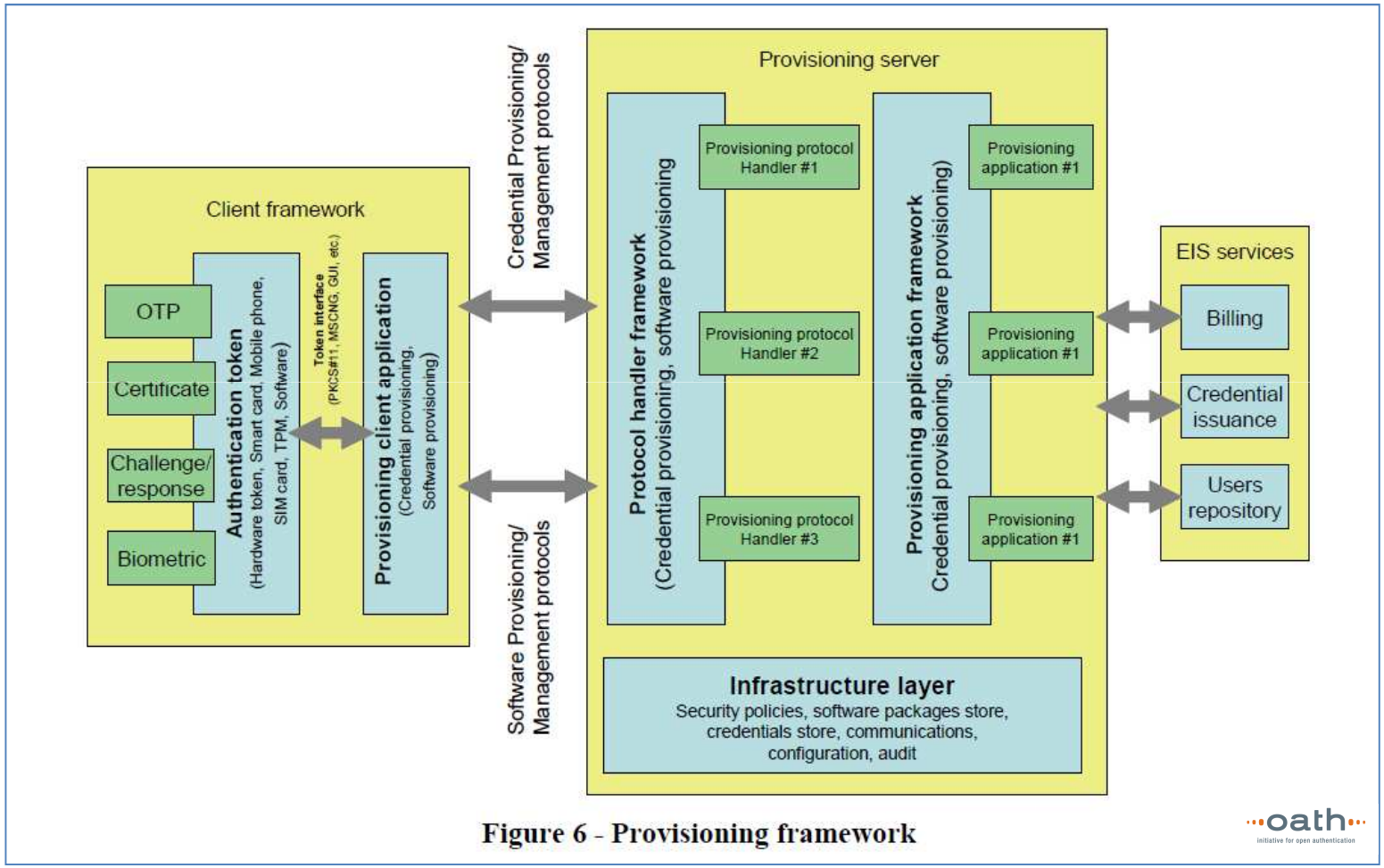


Figure 6 - Provisioning framework

Existing Credential Provisioning Protocols 1/2

XML Key Management Specification - XKMS provides a secure method for the registration and subsequent life-cycle management of public-key information. It comprises two parts: the XML Key Information Service Specification (XKISS) and the XML Key Registration Service Specification (XKRSS). The XKRSS specification defines an interface for a Web service that performs registration of public-key information. Once registered, the public key may be used in conjunction with other Web services, including XKISS. See [XKMS].

PKCS #10, CRMF and PKCS #7 - PKCS #10 and CRMF define standard syntax for certificate requests. Certificate requests are sent to a certification authority, which transforms the request into an X.509 public-key certificate. The resulting certificate, or certificate chain, is usually returned in PKCS #7 format. PKCS #10, CRMF and PKCS #7 are widely supported in public-key infrastructures and public-key enabled applications. See [RFC2986], [RFC2511] and [RFC2315].

Certificate Management Protocol - CMP provides protocols for certificate requests and management. The protocol messages are defined for X.509 certificate creation and management. CMP defines online interactions between PKI components, including an exchange between a certification authority and a client. See [RFC2510].

Certificate Management Messages over CMS - CMC is a certificate management protocol using CMS. The protocol defines an interface to public-key infrastructure components, based on CMS, PKCS #10 and CRMF. See [RFC2797] and [RFC2630].

oath
initiative for open authentication

Existing Credential Provisioning Protocols 2/2

Simple Certificate Enrollment Protocol - SCEP was proposed by Cisco in an Internet draft. Its most compelling feature is the possibility for automatically enrolling certificates for large-scale installations. SCEP supports the RSA public-key algorithm only and leverages PKCS #7. See [SCEP].

Simple Password-authentication Exponential Key Exchange – SPEKE provides authentication and key establishment over an insecure channel using only a small password, without risk of off-line dictionary attack. SPEKE is a variant of Diffie-Hellman Encrypted Key Exchange (DHEKE). See [SPEKE] and [DHEKE].

Crypto Token Key Initialization Protocols (CTKIP) Proposal - The CTKIP proposal provides a secure method of initializing and configuring cryptographic tokens with secret keys, without exposing the secrets to any entities other than the server and the cryptographic token itself. The protocol does not require private-key capabilities in the cryptographic token, and does not mandate an established public-key infrastructure. The initialization session may be secured either using a key, agreed beforehand between the client and the server, or using the server's public key. See [CTKIP].

Dynamic Symmetric Key Provisioning Protocol (DSKPP) Proposal –DSKPP is a proposed standard for key provisioning that is based on the OATH-developed DSKPP Internet Draft and CTKIP. It is the target IETF standard provisioning protocol that OATH sponsors. See [DSKPP] and [DSKPP1].

oath
initiative for open authentication

Software Provisioning Protocols

Browser based download over HTTPS - Software modules can be downloaded to most devices using a standard mobile Internet browser over HTTPS.

Download Over-The-Air (DLOTA) protocol - The DLOTA protocol is defined by the OMA Forum. It defines a protocol for discovering and downloading content and applications to mobile devices. The protocol can be used to download authentication token software modules. DLOTA security relies fully on security in the transport layer, i.e. it uses HTTP basic authentication. See [DLOTA].

Java MIDP OTA provisioning - The Java MIDP 2.0 download protocol allows discovery and delivery of Java MIDLets to Java devices. See [MIDPOTA].

GSM 03.48 applet download - GSM 03.48 defines a secure protocol for over-the-air delivery and subsequent life-cycle management of SIM applets to SIM cards. See [GSM0348].

End Volume 1

Sylvain MARET / @smaret

sylvain.maret@openid.ch
http://www.slideshare.net/smaret
http://www.linkedin.com/in/smaret

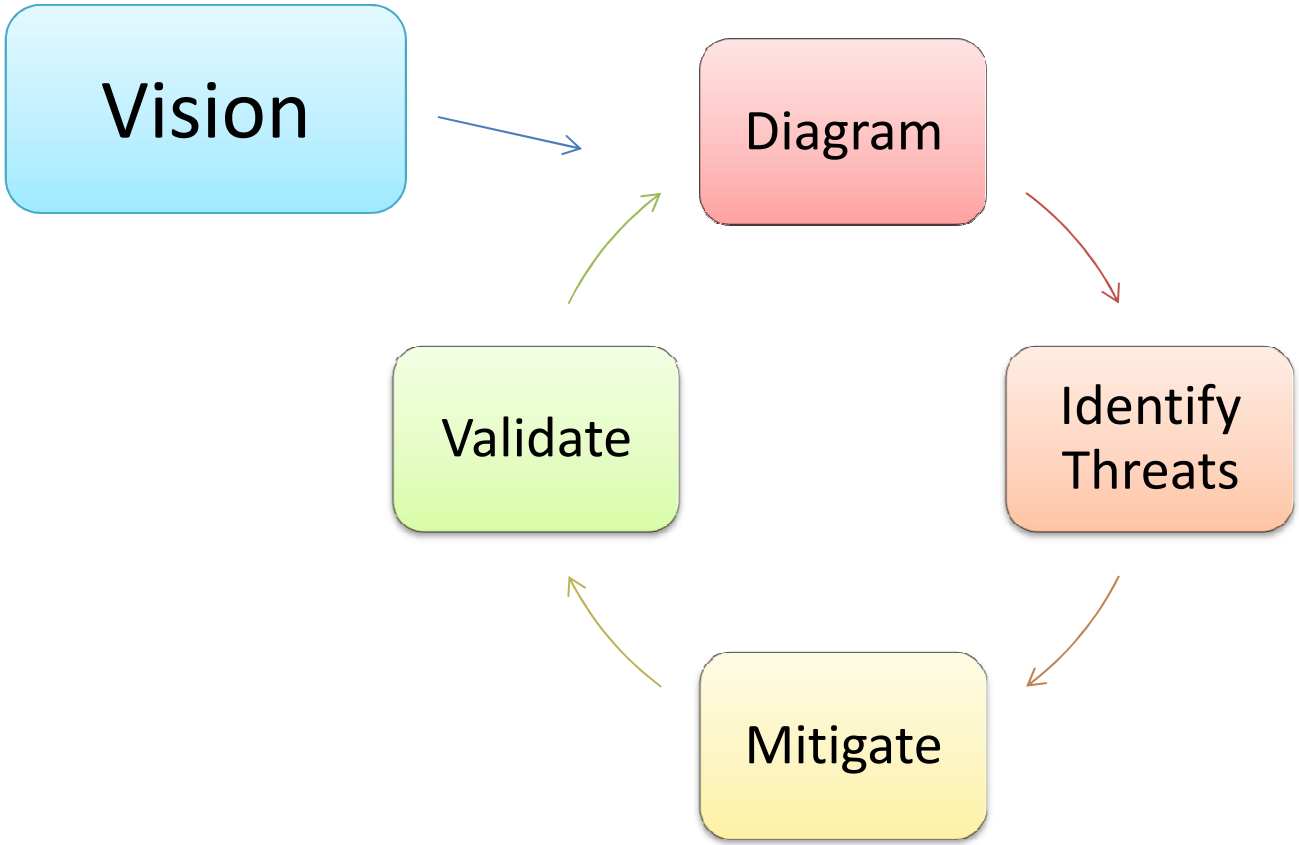
Appendices

Threat Modeling

DFD

STRIDE

Threat Modeling Process



DFD symbols

External Entity

The external entity shape is used to represent any entity outside the application that interacts with the application via an entry point.

Process

The process shape represents a task that handles data within the application. The task may process the data or perform an action based on the data.

DFD Symbols

Multiple Process

The multiple process shape is used to present a collection of subprocesses. The multiple process can be broken down into its subprocesses in another DFD.

Data Store

The data store shape is used to represent locations where data is stored. Data stores do not modify the data, they only store data.

DFD Symbols

Data Flow

The data flow shape represents data movement within the application. The direction of the data movement is represented by the arrow.

Data Flow

Privilege Boundary

The privilege boundary shape is used to represent the change of privilege levels as the data flows through the application.

Privilege Boundary

Trust boundaries that intersect data flows

- Points/surfaces where an attacker can interject
  - Machine boundaries, privilege boundaries, integrity boundaries are examples of trust boundaries
  - Threads in a native process are often inside a trust boundary, because they share the same privs, rights, identifiers and access
- Processes talking across a network always have a trust boundary

DFD Level

- Level 0 - Context Diagram
  - Very high-level; entire component / product / system
- Level 1 Diagram
  - High level; single feature / scenario
- Level 2 Diagram
  - Low level; detailed sub-components of features
- Level 3 Diagram
  - More detailed
  - Rare to need more layers, except in huge projects or when you're drawing more trust boundaries

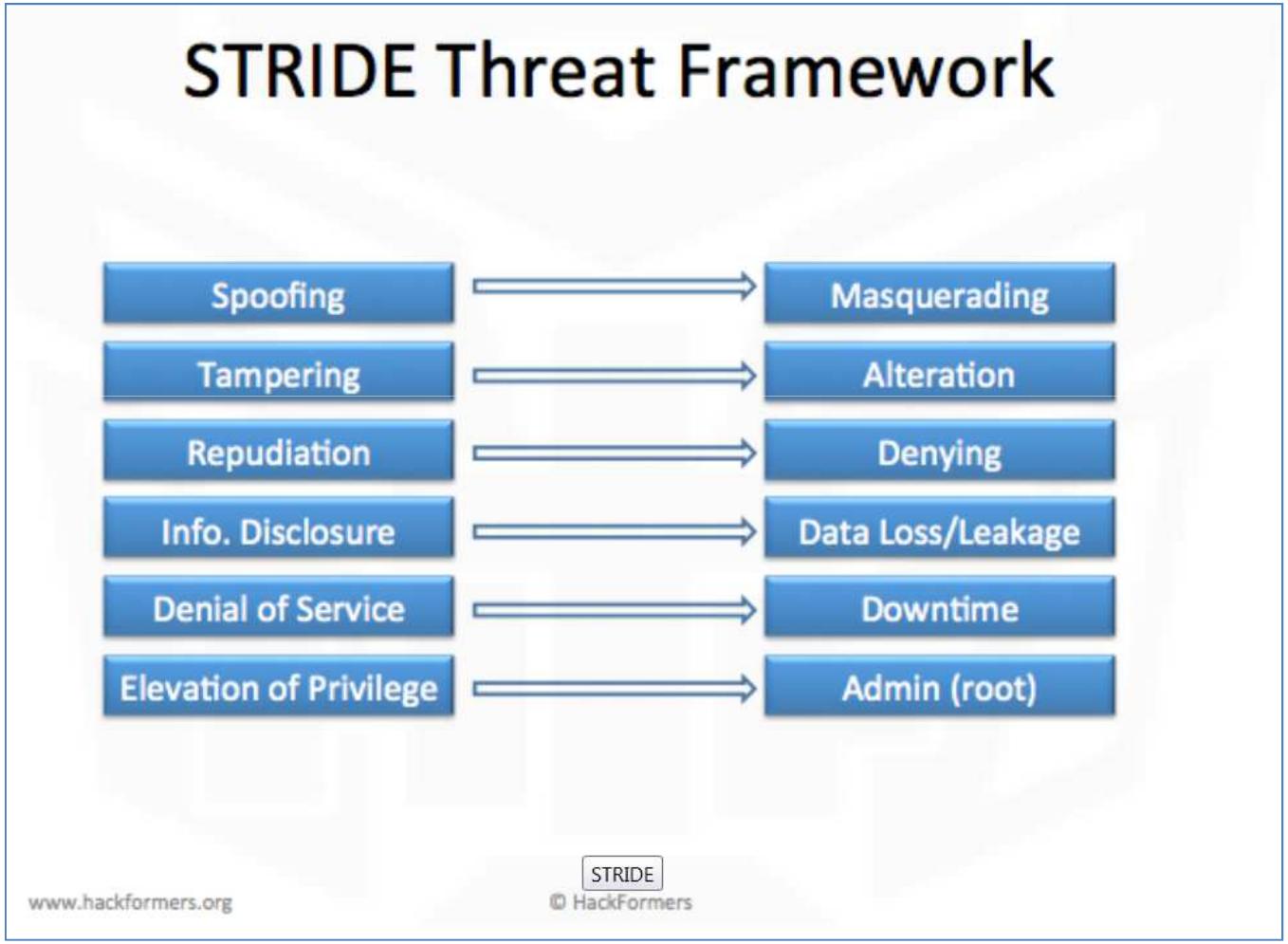
STRIDE - Tool

Threat	Property	Definition	Example
Spoofing	Authentication	Impersonating something or someone else.	Pretending to be any of billg, xbox.com or a system update
Tampering	Integrity	Modifying data or code	Modifying a game config file on disk, or a packet as it traverses the network
Repudiation	Non-repudiation	Claiming to have not performed an action	"I didn't cheat!"
Information Disclosure	Confidentiality	Exposing information to someone not authorized to see it	Reading key material from an app
Denial of Service	Availability	Deny or degrade service to users	Crashing the web site, sending a packet and absorbing seconds of CPU time, or routing packets into a black hole
Elevation of Privilege	Authorization	Gain capabilities without proper authorization	Allowing a remote internet user to run commands is the classic example, but running kernel code from lower trust levels is also EoP

STRIDE – Security Controls

STRIDE Threat List		
Type	**Examples**	**Security Control**
Spoofing	Threat action aimed to illegally access and use another user's credentials, such as username and password.	Authentication
Tampering	Threat action aimed to maliciously change/modify persistent data, such as persistent data in a database, and the alteration of data in transit between two computers over an open network, such as the Internet.	Integrity
Repudiation	Threat action aimed to perform illegal operations in a system that lacks the ability to trace the prohibited operations.	Non-Repudiation
Information disclosure	Threat action to read a file that one was not granted access to, or to read data in transit.	Confidentiality
Denial of service	Threat aimed to deny access to valid users, such as by making a web server temporarily unavailable or unusable.	Availability
Elevation of privilege	Threat aimed to gain privileged access to resources for gaining unauthorized access to information or to compromise a system.	Authorization

SRIDE



SRIDE

Figure 2 Security Properties

Property	Description
Confidentiality	Data is only available to the people intended to access it.
Integrity	Data and system resources are only changed in appropriate ways by appropriate people.
Availability	Systems are ready when needed and perform acceptably.
Authentication	The identity of users is established (or you're willing to accept anonymous users).
Authorization	Users are explicitly allowed or denied access to resources.
Nonrepudiation	Users can't perform an action and later deny performing it.

Figure 3 Threats and Security Properties

Threat	Security Property
Spoofing	Authentication
Tampering	Integrity
Repudiation	Non-repudiation
Information disclosure	Confidentiality
Denial of service	Availability
Elevation of privilege	Authorization

STRIDE Threat & Mitigation Techniques List

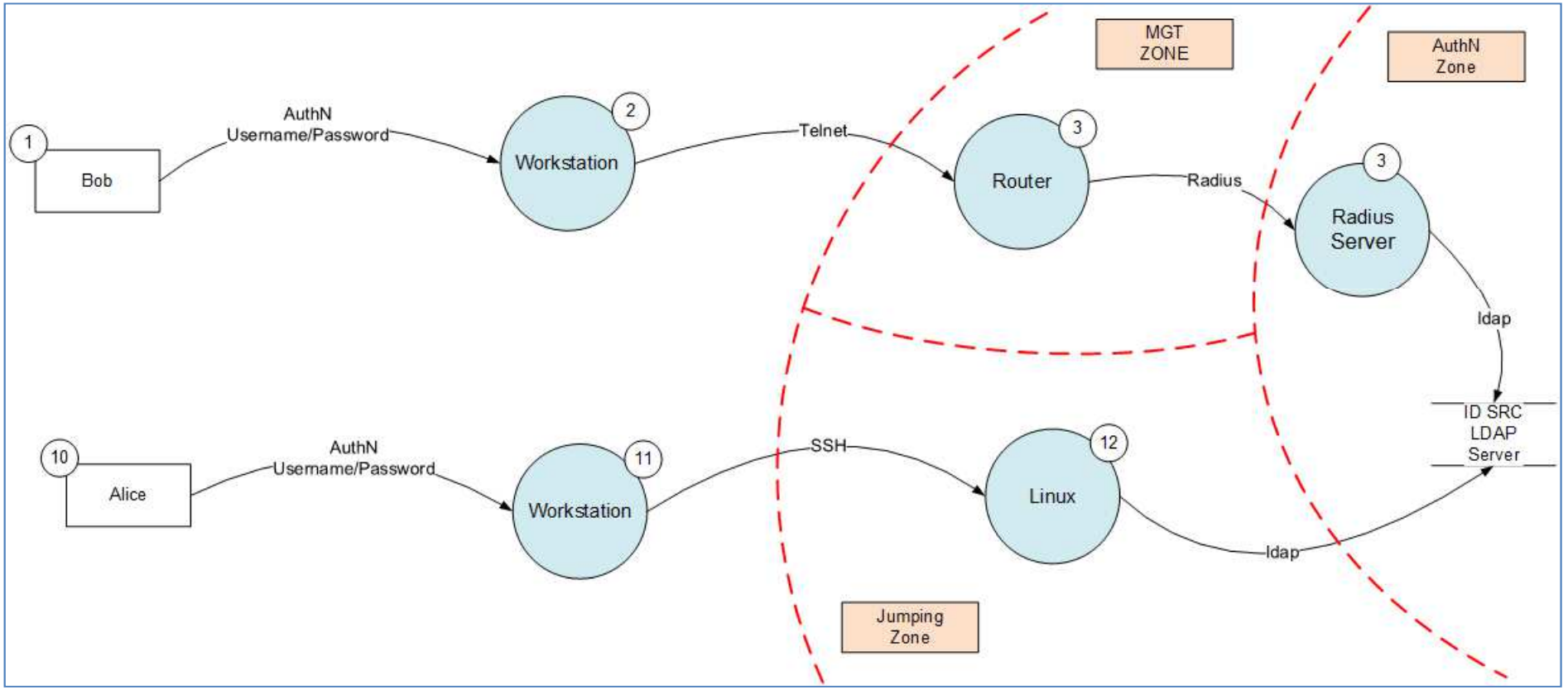
Threat Type	Mitigation Techniques
Spoofing Identity	1. Appropriate authentication 2. Protect secret data 3. Don't store secrets
Tampering with data	1. Appropriate authorization 2. Hashes 3. MACs 4. Digital signatures 5. Tamper resistant protocols
Repudiation	1. Digital signatures 2. Timestamps 3. Audit trails
Information Disclosure	1. Authorization 2. Privacy-enhanced protocols 3. Encryption 4. Protect secrets 5. Don't store secrets
Denial of Service	1. Appropriate authentication 2. Appropriate authorization 3. Filtering 4. Throttling 5. Quality of service
Elevation of privilege	1. Run with least privilege

DFD & STRIDE

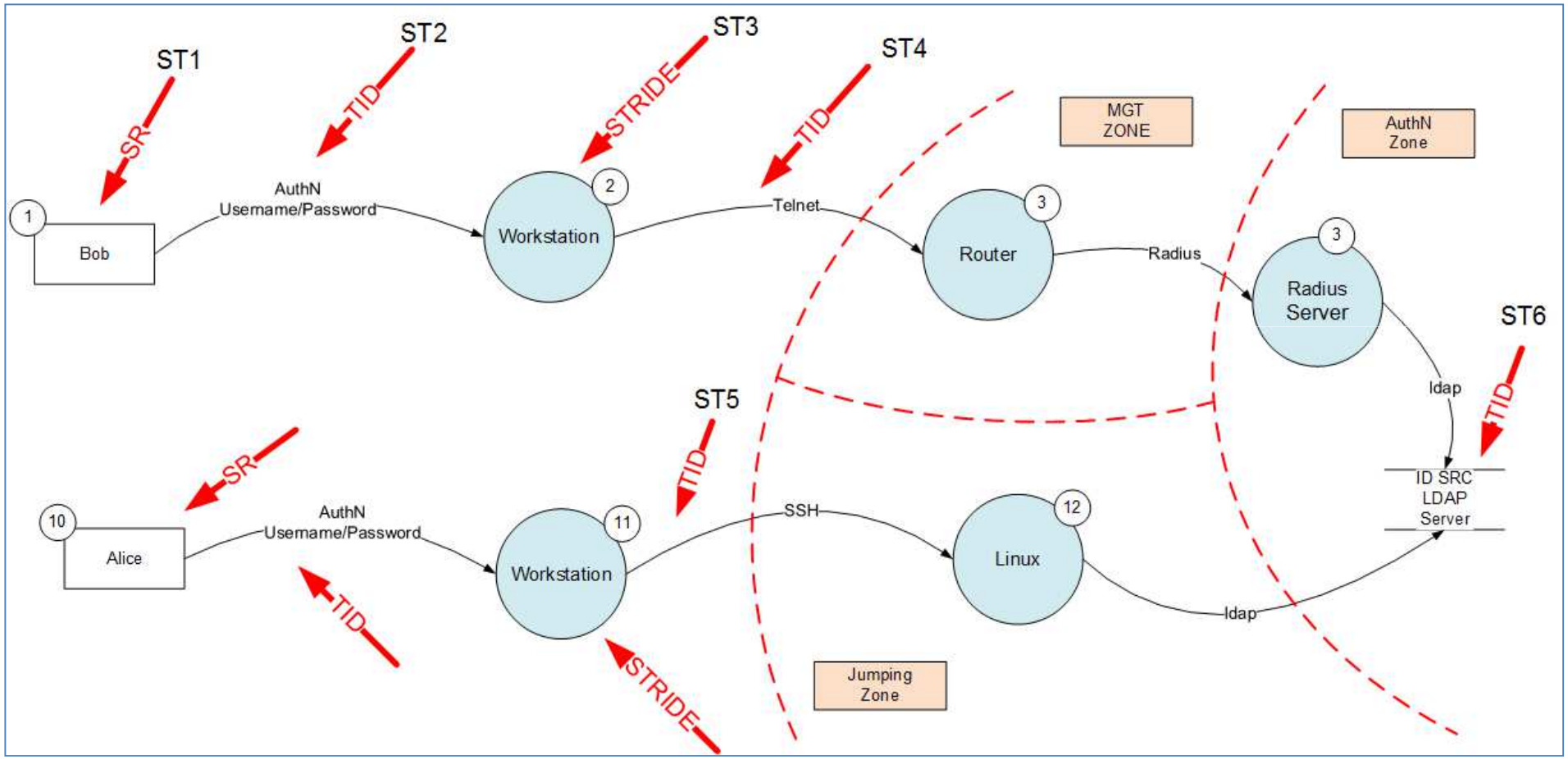
Figure 5 Threats Affecting Elements

Element	Spoofing	Tampering	Repudiation	Information Disclosure	Denial of Service	Elevation of Privilege
Data Flows		X		X	X	
Data Stores		X		X	X	
Processes	X	X	X	X	X	X
Interactors	X		X			

DFD AuthN 1FA



DFD – AuthN 1FA / STRIDE



HSPD-12

PIV AuthN

Homeland Security Presidential Directive/Hspd-12

the White House
President George W. Bush

For Immediate Release
Office of the Press Secretary
August 27, 2004

Homeland Security Presidential Directive/Hspd-12

Subject: Policy for a Common Identification Standard for Federal Employees and Contractors

http://www.dhs.gov/homeland-security-presidential-directive-12

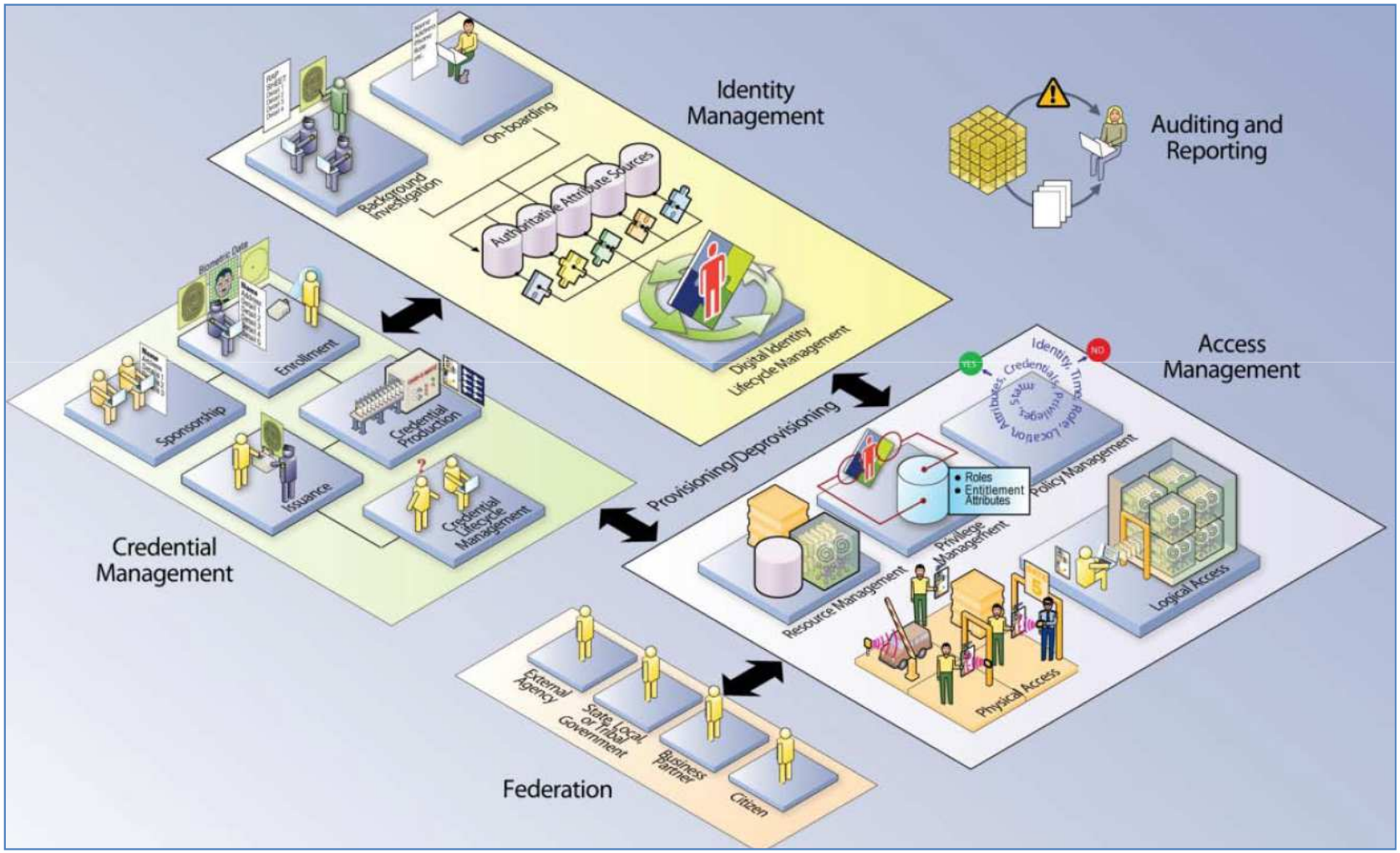
FIPS 201 / PIV

- Federal Information Processing Standard 201, Personal Identity Verification (PIV) of Federal Employees and Contractors, March 2006.
  - (See http://csrc.nist.gov)

- FIPS 201 (Federal Information Processing Standard Publication 201) is a United States federal government standard that specifies Personal Identity Verification (PIV) requirements for Federal employees and contractors.

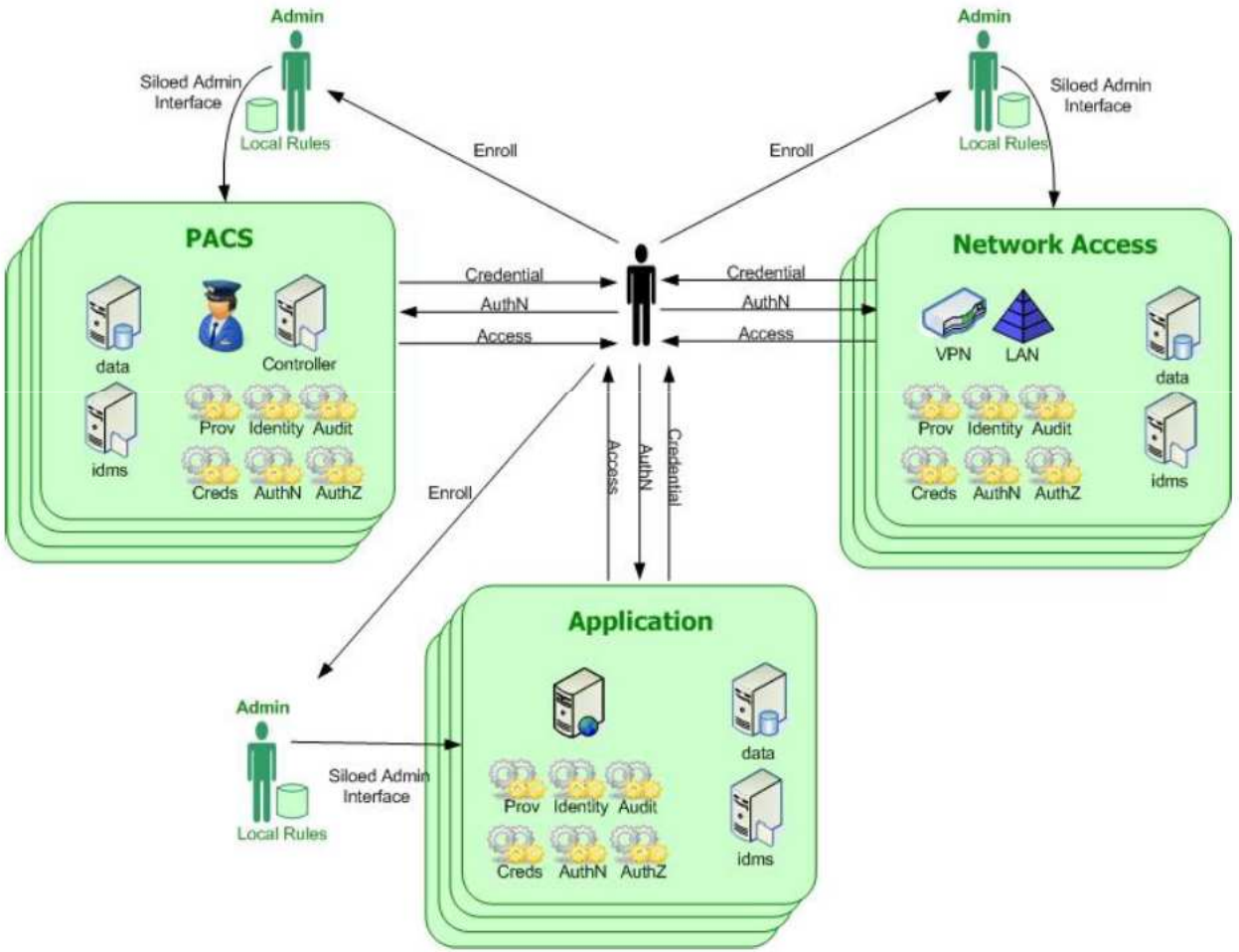
- http://www.idmanagement.gov/

FICAM Roadmap



FICAM Roadmap and Implementation Guidance
Version 1.0

FICAM
Identity, Credential, & Access Management



FICAM Roadmap and Implementation Guidance
Version 1.0

FICAM
Identity, Credential, & Access Management

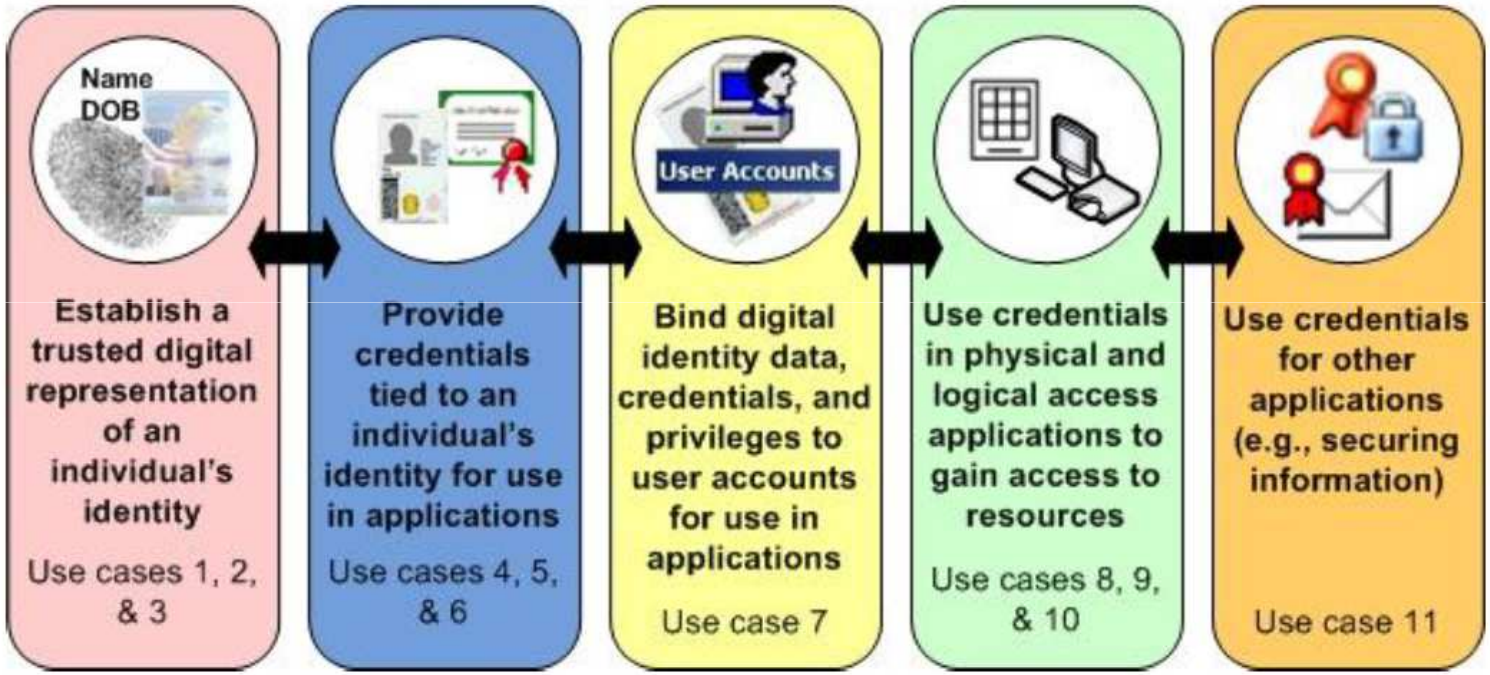


Figure 14: Use Case Functional Overview

LOA

PIV Assurance Levels	Comparable OMB E-Authentication Levels	
	Level Number	Description
LITTLE or NO confidence	Level 1	Little or no confidence in the asserted identity's validity
SOME confidence	Level 2	Some confidence in the asserted identity's validity
HIGH confidence	Level 3	High confidence in the asserted identity's validity
VERY HIGH confidence	Level 4	Very high confidence in the asserted identity's validity

http://www.idmanagement.gov/

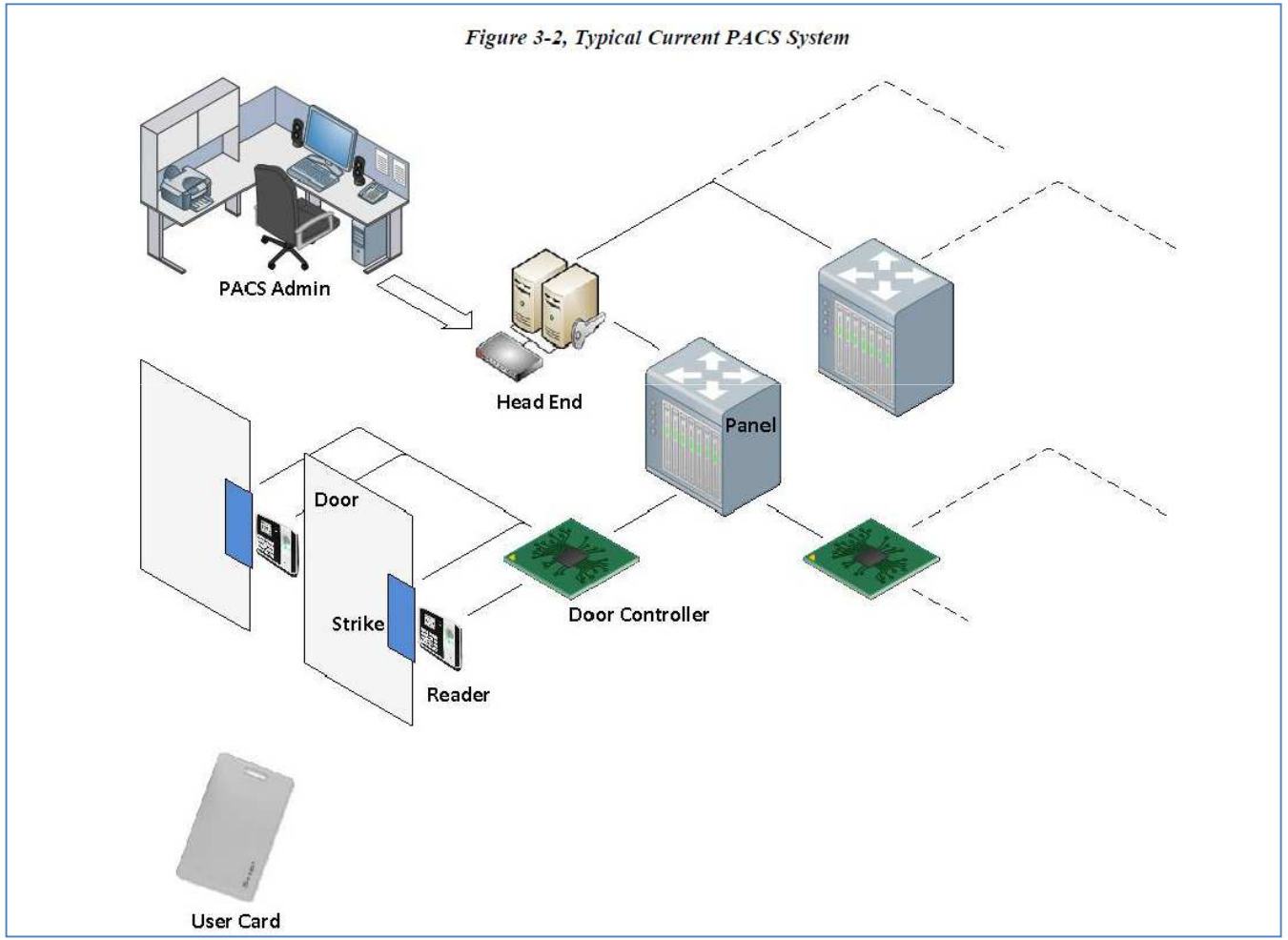
LOA

Policy Foundation: NIST Special Pub 800-63

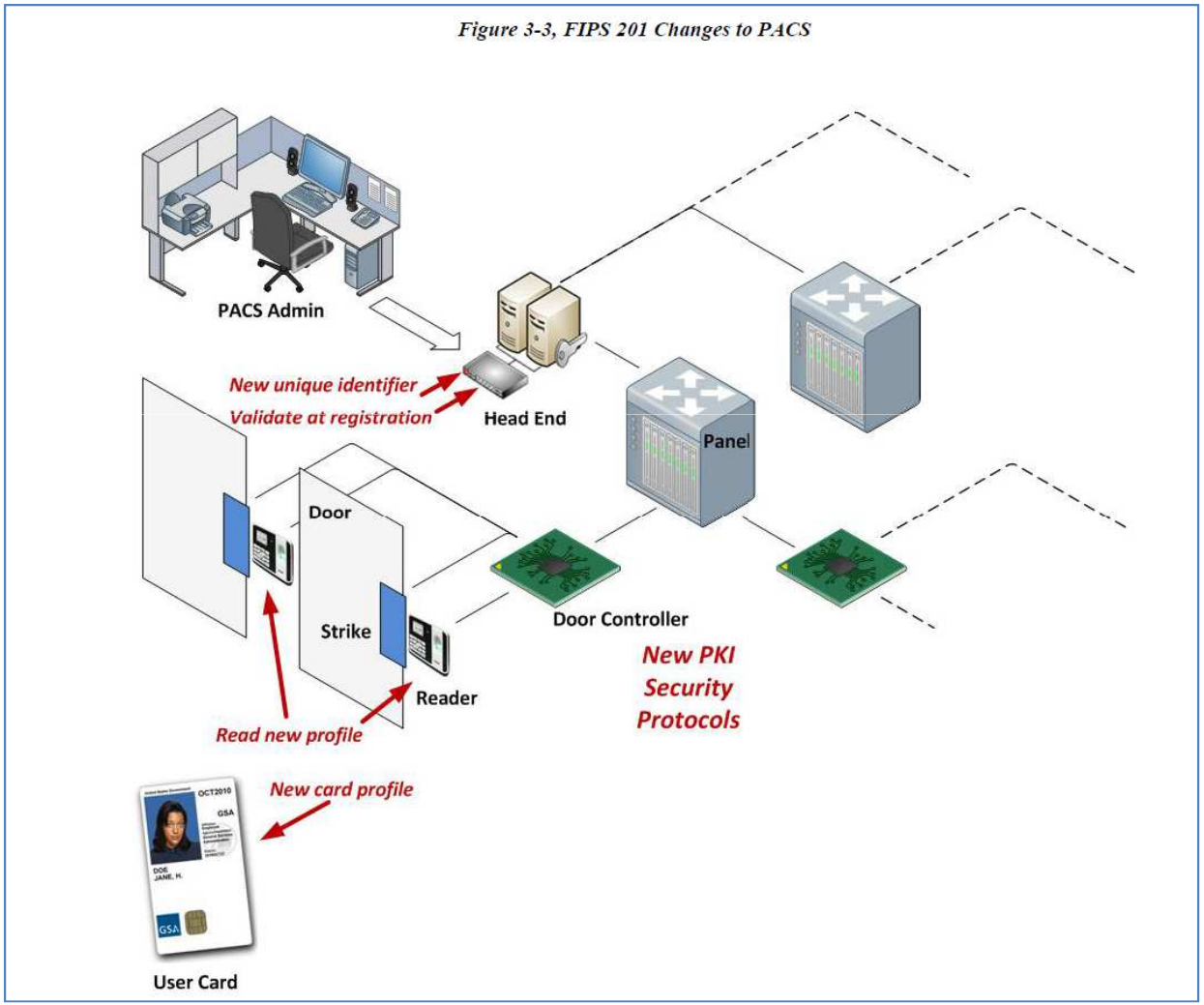
- SP 800-63 Technical Guidance

Allowed Token Types	Assurance Level 1	2	3	4
Hard crypto token	√	√	√	√
One-time Password Device	√	√	√	
Soft crypto token	√	√	√	
Password & PINs	√	√		

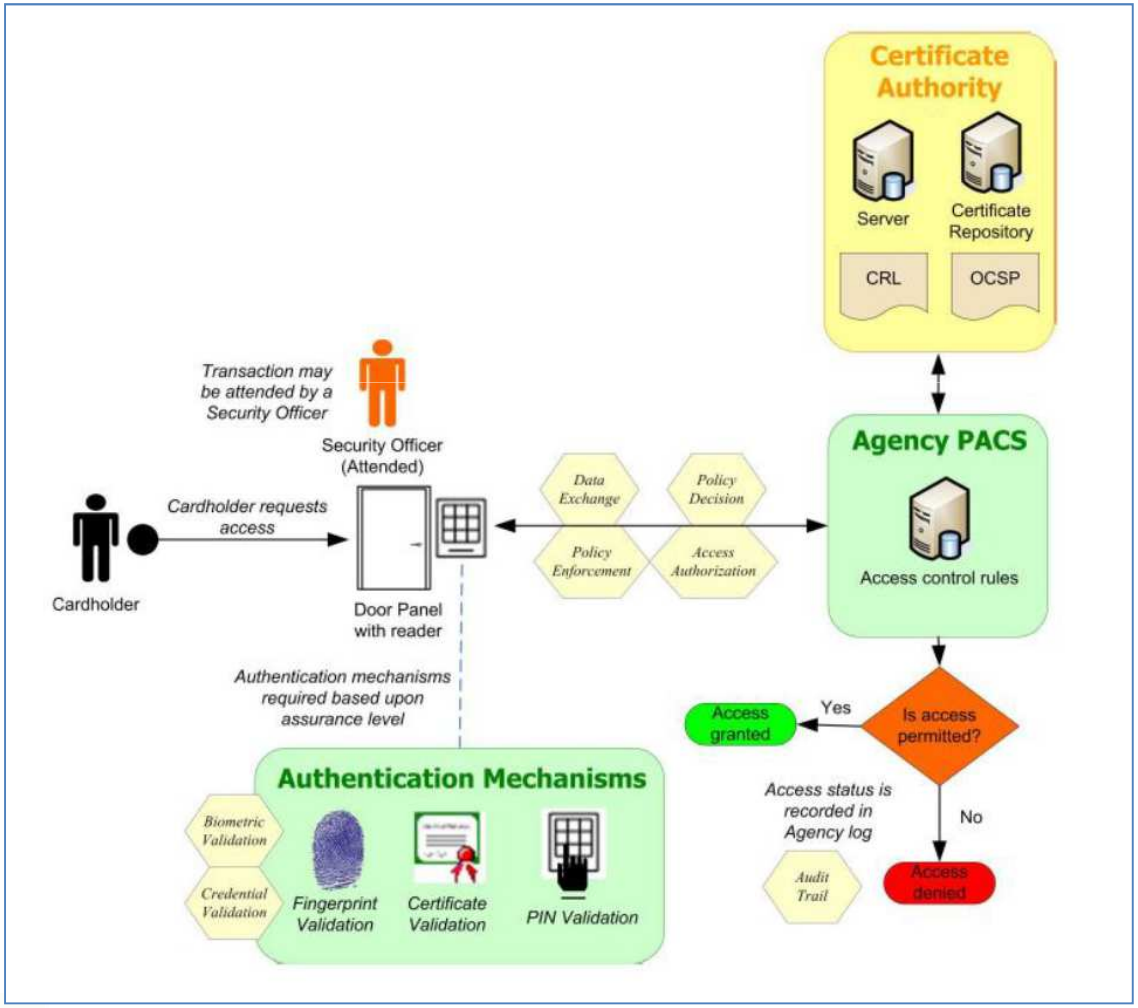
FICAM Roadmap - PACS



FICAM Roadmap - PACS



FICAM Roadmap



PIV Card & Reader

PIVMAN – FIPS 201