Segurança de Sistemas e dados (MSI 2021/2022)

Aula 2

Rolando Martins
DCC – FCUP
Slides Adaptados do Prof. Manuel Eduardo Correia

Biometric Accuracy

* never get identical templates
* problems of false match / false non-match

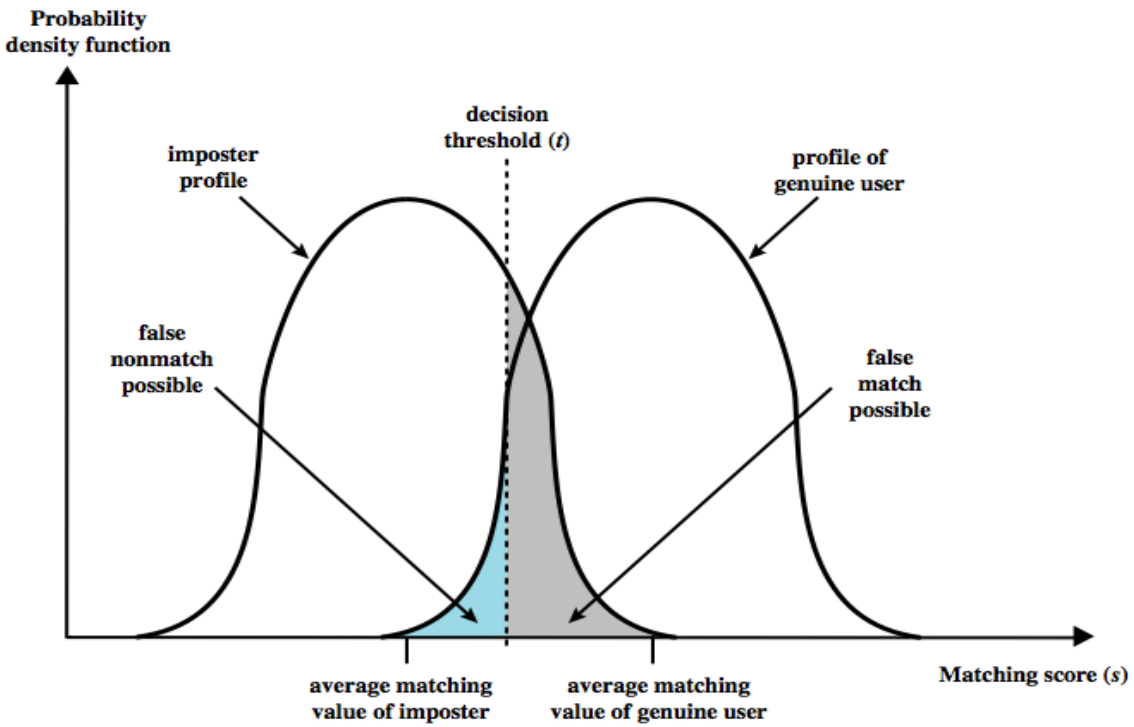


Figure 3.8 Profiles of a biometric characteristic of an imposter and an authorized users. In this depiction, the comparison between presented feature and a reference feature is reduced to a single numeric value. If the input value (s) is greater than a preassigned threshold (t), a match is declared.

Biometric Errors

* Fraud rate versus insult rate
  * **Fraud**: Trudy mis-authenticated as Alice **(False Positive (FP))**
  * **Insult**: Alice not authenticated as Alice **(False Negative (FN))**
* For any biometric, can decrease fraud or insult, but other one will increase
* For example
  * 99% voiceprint match: low fraud, high insult
  * 30% voiceprint match: high fraud, low insult
* **Equal error rate: rate where fraud == insult**
  * A way to compare different biometrics

Biometric Accuracy

* can plot characteristic curve
* pick threshold balancing error rates

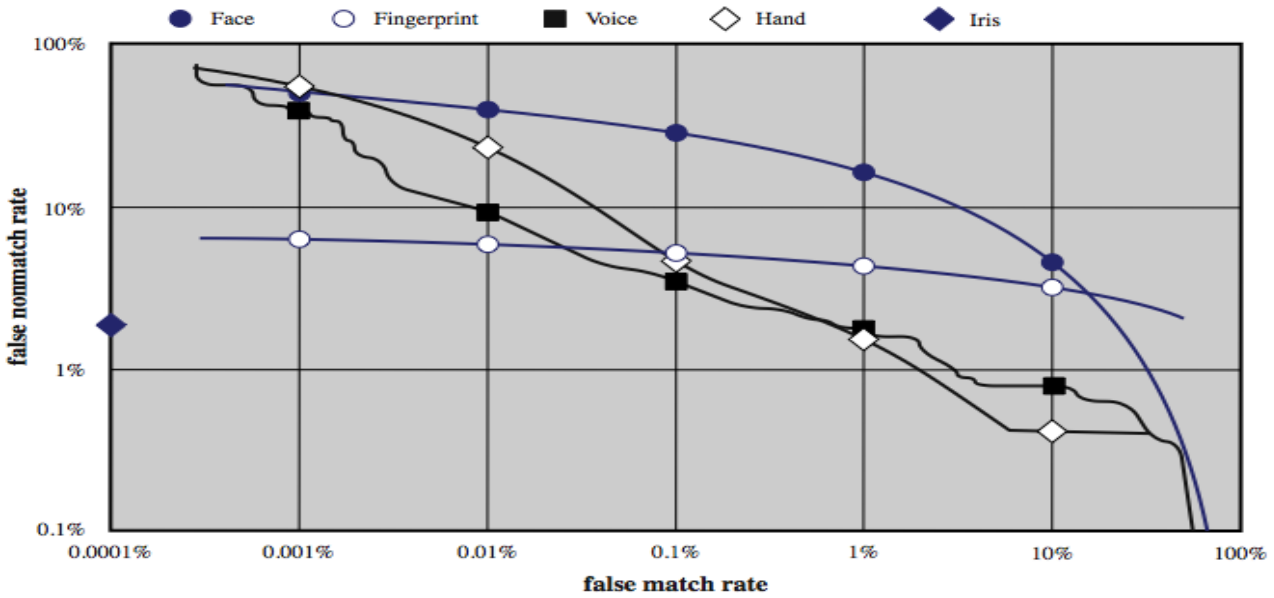


Figure 3.10 Actual biometric measurement operating characteristic curves, reported in [MANS01]. To clarify differences among systems, a log-log scale is used.

Enrollment vs Recognition

* Enrollment phase
  * Subject's biometric info put into database
  * Must carefully measure the required info
  * OK if slow and repeated measurement needed
  * Must be very precise
  * May be weak point of many biometric
* Recognition phase
  * Biometric detection, when used in practice
  * Must be quick and simple
  * But must be reasonably accurate

Operation of a Biometric System

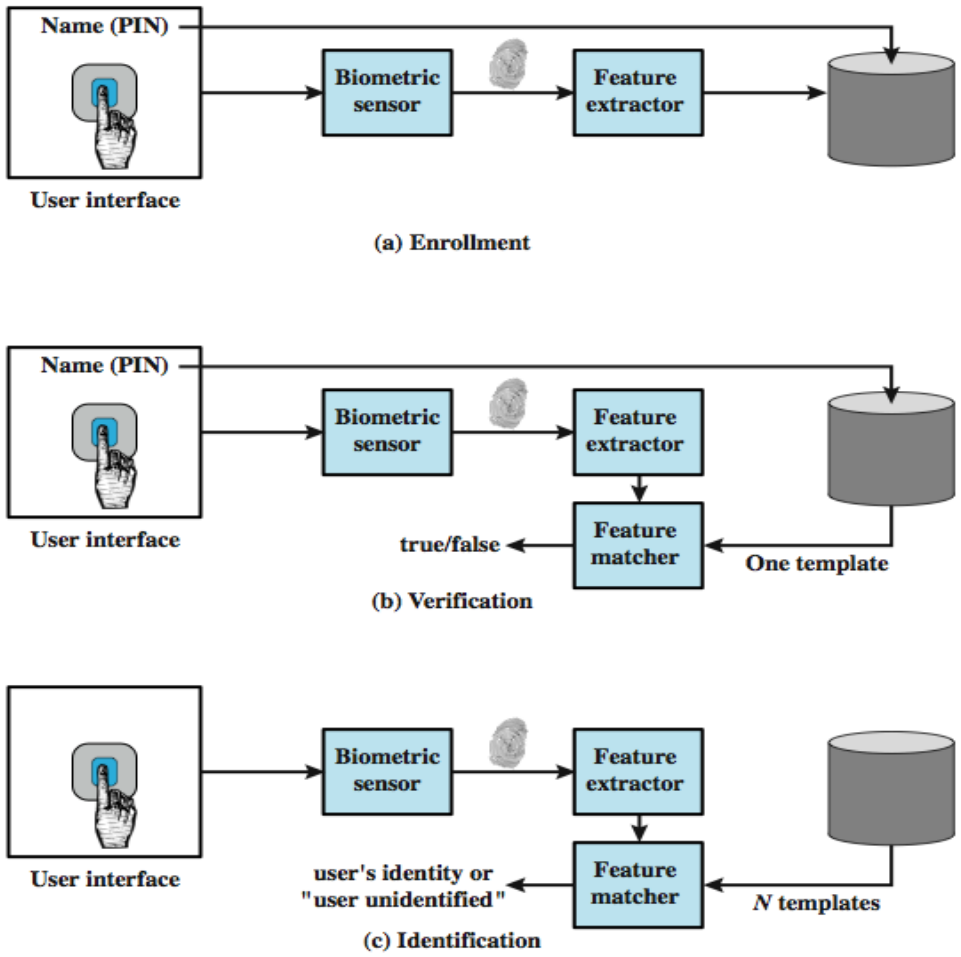


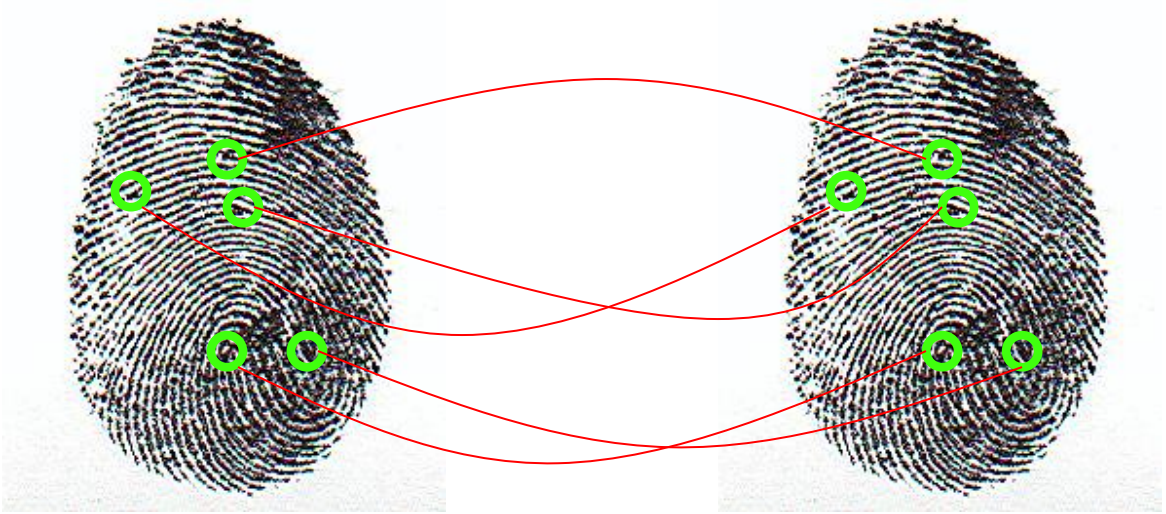
Figure 3.7 A generic biometric system. Enrollment creates an association between a user and the user's biometric characteristics. Depending on the application, user authentication either involves verifying that a claimed user is the actual user or identifying an unkown user.

Fingerprint: Enrollment

* Capture image of fingerprint
* Enhance image
* Identify points

Fingerprint: Recognition

* Extracted points are compared with information stored in a database using statistical matching.



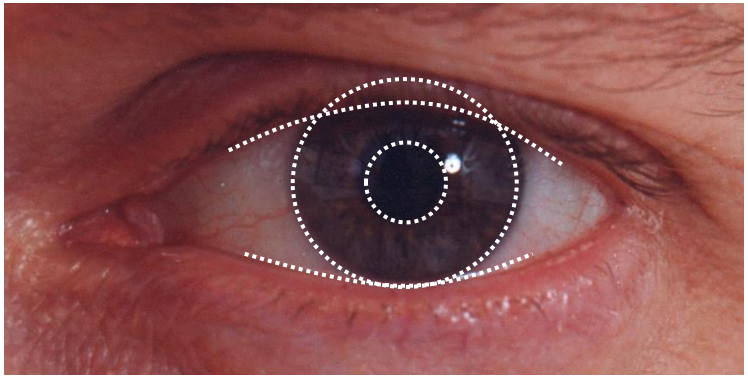
Hand Geometry

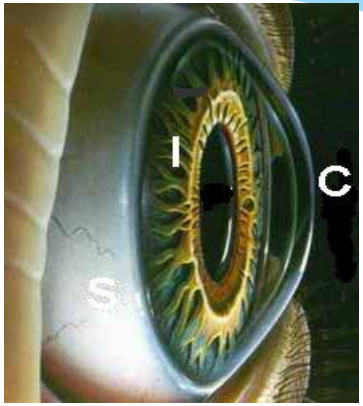
- A popular biometric
- Measures shape of hand
  - Width of hand, fingers
  - Length of fingers, etc.
- Human hands not unique
- Hand geometry sufficient for many situations
- OK for authentication
- Not useful for ID problem

Hand Geometry

* Advantages 
  * Quick — 1 minute for enrollment, 5 seconds for recognition
  * Hands are symmetric — so what?
* Disadvantages 
  * Cannot use on very young or very old
  * Relatively high equal error rate

Iris Patterns





* Iris pattern development is “chaotic”
* Little or no genetic influence
* Different even for identical twins
* Pattern is stable through lifetime

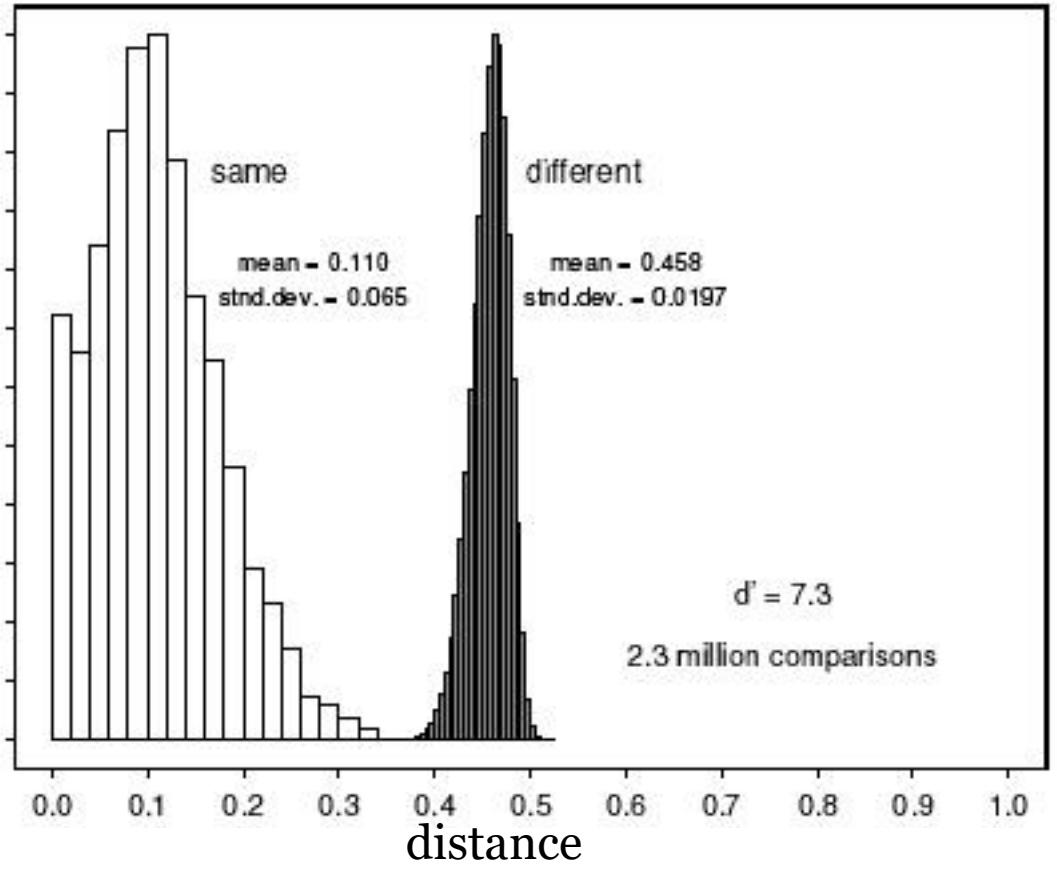
Iris Scan Error Rate

$$distance = \frac{\#bits\ that\ don't\ match}{\#bits\ compared}$$

distance	Fraud rate
0.29	1 in $1.3*10^{10}$
0.30	1 in $1.5*10^{9}$
0.31	1 in $1.8*10^{8}$
0.32	1 in $2.6*10^{7}$
0.33	1 in $4.0*10^{6}$
0.34	1 in $6.9*10^{5}$
0.35	1 in $1.3*10^{5}$


equal error rate


usually used



Equal Error Rate Comparison

* **Equal error rate (EER): fraud == insult rate**
* **Fingerprint** biometric has **EER of about 5%**
* **Hand geometry** has **EER of about 10-3**
* **In theory, iris scan has EER of about 10-6**
  * But in practice, may be hard to achieve
  * Enrollment phase must be extremely accurate
* **Most biometrics much worse than fingerprint!**
* **Biometrics useful for authentication…**
  * **…but identification biometrics almost useless today**

Biometrics: The Bottom Line

* Biometrics are hard to forge
* But attacker could
  * Steal Alice's thumb
  * Photocopy Bob's fingerprint, eye, etc.
  * Subvert software, database, "trusted path" …
* And how to revoke a "broken" biometric?
* **Biometrics are not foolproof**
* Biometric use is limited today
* That should change in the (near?) future

Some devices

Something you have

Tokens

Token Authentication

* object user possesses to authenticate, e.g.
  * embossed card
  * Bank random matrix codes plastic cards
  * magnetic stripe card
  * memory card
  * Smartcard
  * Smartphone App

Should be hard to copy and/or well protected.

Memory Card

* store but do not process data
* magnetic stripe card, e.g. old bank cards
* electronic memory card
  * Old phone cards
* used alone for physical access
* with password/PIN for computer use
* drawbacks of memory cards include:
  * need special reader
  * loss of token issues
  * user dissatisfaction

Smartcard

* credit-card like
* has own processor, memory, I/O ports
  * wired or wireless access by reader
  * may have crypto co-processor
  * ROM, EEPROM, RAM memory
* executes protocol to authentic reader/computer
* also have USB dongles

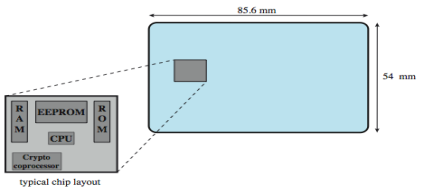


Figure 3.4 Smartcard dimensions. The smartcard chip is embedded into the plastic card and is not visible. The dimensions conform to ISO standard 7816-2

Cartão cidadão

* BI
* Segurança Social
* Serviço Nacional de Saúde
* Número de contribuinte
* Cartão de Eleitor
* Documento de viagem.

2-factor Authentication

* Requires any 2 out of 3 of
  - Something you know
  - Something you have
  - Something you are
* Examples
  * ATM: Card and PIN
  * Credit card: Card and signature
  * Password generator: Device and PIN
  * Smartcard with password/PIN

Problems with Passwords Revisited

- Selection
  - Good secure passwords are hard to find
- Memorization
  - It is easy to forget infrequently used passwords
  - It is hard to remember secure passwords
- Reuse
  - To many different passwords to memorize
  - People reuse the same password all over

Problems with Passwords in clinical settings

* In hospitals it is common to find post-it notes with application passwords.
* Health professionals often share accounts with each other.
  * To cover each other backs.
  * To circumvent hardly usable security
  * To delegate tasks to interns.
  * To save lives. (badly designed RBAC models and hard to use security mechanisms)
* Keyboards are also among the dirtiest objects in the planet

Problems with Passwords Revisited

- Sharing
  - When working in groups it is common for people to share passwords
  - Non repudiation is easy to circumvent with password sharing
  - When people do not assess the real value of their digital assets they share passwords.
- Malware (Virus, Trojans)
  - Password Sniffing, Phishing Attacks, etc...

Problems with Passwords Revisited

- Online Banking is one prime example of management of highly valuable assets on the Internet.
  - Online Banks are very convenient for the costumer and save of lot money to the Banks.
  - “Phishing” attacks became widespread and are quite effective at stealing user credentials.
  - Banking dedicated Malware provides high returns to the attacker
- Risk Analysis tell us that Login/Password is not appropriate to protect these assets.

Classical Factor Two Authentication tokens

- The classic solution has been to employ factor two (difficult to clone) authentication

Classical Factor Two Authentication tokens

* Users need to carry with them additional physical tokens and/or readers.
* Generally, requires specific drivers and middleware software installed on users' workstations.
* Are proprietary in Nature and incompatible with each other.
  * Single purpose (Generally you cannot use Institution A token to authenticate yourself into Institution B).

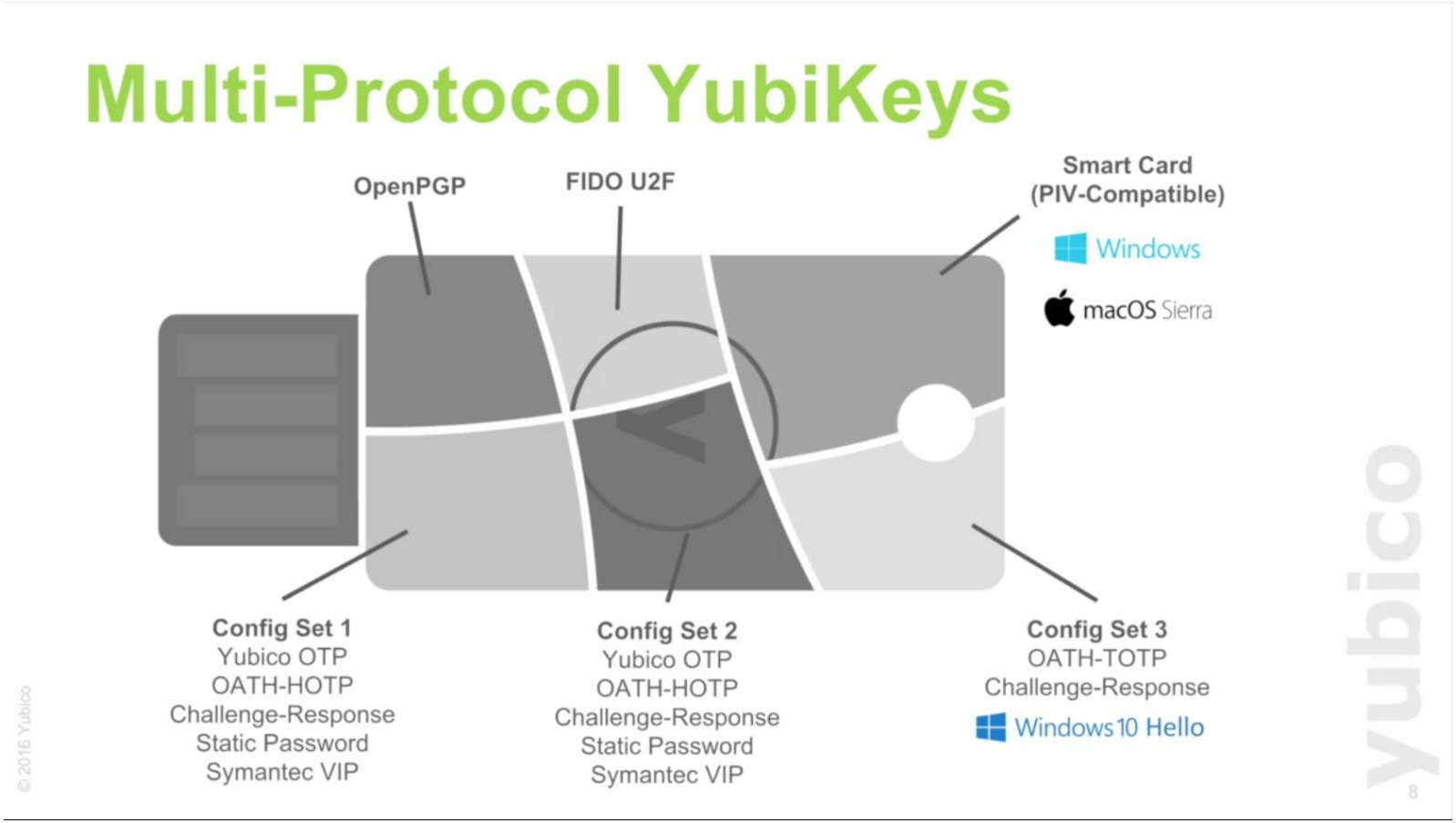
More Flexible Factor Two Authentication tokens

* Yubikeys:
  * Low cost one time password (OTP) generator token (40 chars).
  * Connects to USB port.
  * Acts like a keyboard, no driver required.
  * Press button with your finger to generate a new OTP.
  * Very easy to integrate with legacy login/password authentication schemes.
  * Token validation service on the cloud.
  * Widely deployed by well know Internet companies (Paypal, Google, LastPass,…)
  * Direct support currently being integrated into Google chrome for a more seamless authentication with HTML5 sites.

More Flexible Factor Two Authentication tokens

* Yubikeys:
  * Current Version has two slots. Each can store 128bits.
  * These can be used in several modes:
    * Yubikey OTP
    * OAUTH-HOTP (RFC 4226)
    * Static 128 bit password
    * Challenge Response
  * Newer models are NFC enabled.
    * Yubikey NEO
  * Yubikey software is mostly open Source.
  * For the server side Yubico also supply a low cost HSM to securely protect shared secrets.

Newer Generation YubiKeys



Other Branded MultiApp USB Tokens

https://www.ftsafe.com/products/FIDO/NFC

* USB and NFC communications
* FIDO U2F, OATH HOTP
* GIDs (Generic Identity Device Specification) and Windows Hello
* Java smartcard

* Supported in Android, Windows, Linux and MacOs

Ubiquitous Authentication tokens ?

- Classical Authentication tokens are very expensive do deploy on a large scale.
- Are difficult to use by the targeted regular users.
- There is however another popular device that can be used as a factor 2 authentication token and:
  - Nowadays it is as common as House Keys:

The Mobile Phone

Using your Mobile Phone as na Authentication token

- Everyone has a mobile phone.
- Every phone has SMS capabilities.
  - The cell phone authentication infrastructure is reasonably secure.
- We can use a SMS message as a side channel to share a temporary secret that only the possessor of a mobile phone can see.
  - The Online banking industry was one of the first to use SMS messaging as a means to authenticate critical operations.
  - Nowadays it is almost impossible use online banking services without a registered phone.

Using your mobile smart Phone/Apps as an Authentication token

- SMS messaging has a cost.
- It can be substantially expensive if you are abroad.
- It could not work if messaging takes too much time (60 seconds delay).
- It is cumbersome to use.
- It is not safe as it used to be:
  - **Sting Ray Devices:** When operating in active mode, the Stingray device mimics a wireless carrier cell tower in order to force all nearby mobile phones and other cellular data devices to connect to it. (https://en.wikipedia.org/wiki/Stingray_phone_tracker)
  - **Signaling Systems n° 7 (SS7) Vulnerabilities**: dates back to the 1970s - "*89% of subscribers' SMS can be intercepted; 58% of subscribers can be tracked, and half of all phone calls can be wiretapped*" (https://secure-voice.com/ss7_attacks)
- Nowadays the mobile phone can do much more then just "texting".
- We can do so much more !

Using your Phone as a secure Authentication token

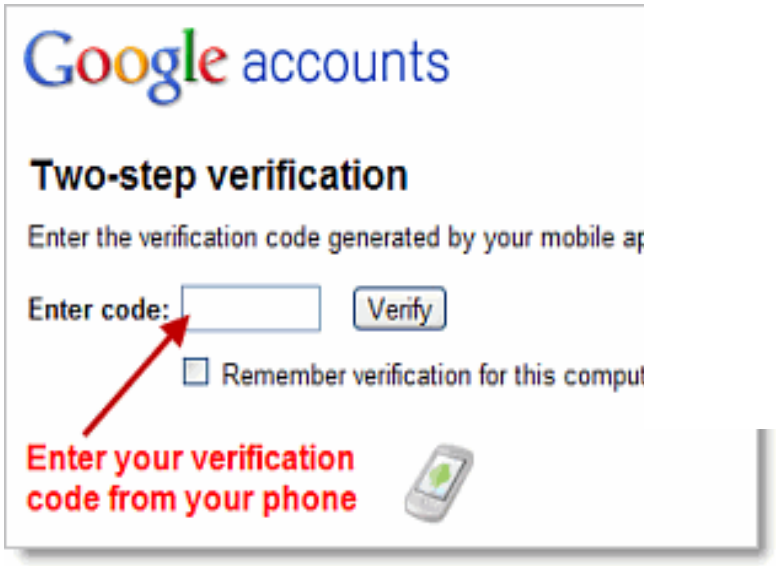
- Smart phones run APPs
- Recent Android phones come with their own Secure Element (SE) Built-in .
  - Google Nexus – Google Wallet; Myfare emulation.
- Near Field Communication (NFC) and card emulation capacity is becoming common place in the more recent Android Devices
  - It is now possible to emulate smart cards at the application level.
  - Combine this with the (SE) and we open a whole new set of potential applications .
    - Physically access (Myfare Locks); Innovative mobile payment systems; Transportation cards; Loyalty cards; etc…
  - All cards in our wallets could all be securely integrated into our NFC enabled smartphone equipped with a SE element.

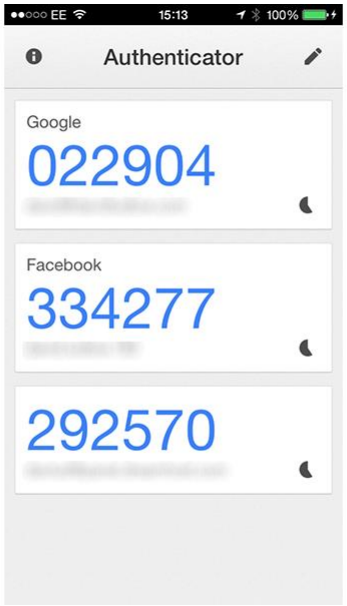
Using your smart Phone/Apps as an Authentication token

- Google had a serious problem with authentication.
  - Solely based on login/password
  - Highly vulnerable to MITM attacks for credentials harvesting as attested by the chinese incident of 2009/2010 – Operation Aurora (https://en.wikipedia.org/wiki/Operation_Aurora)
- You can secure your google account with an **APP acting as a factor2 authenticator (Google Authenticator)** implementation of one-time passcode generators for several mobile platforms.
  - Support the HMAC-Based One-time Password (HOTP) algorithm specified in RFC 4226
  - Time-based One-time Password (TOTP) algorithm
  - https://github.com/google/google-authenticator

Using your smart Phone/Apps as an Authentication token

- With the Google Authenticator you can declare a certain browser at a certain computer to be trustable.
- All the others will require the proof of possession of your mobile device for the login to succeed.





Using your smart Phone/Apps as an Authentication token

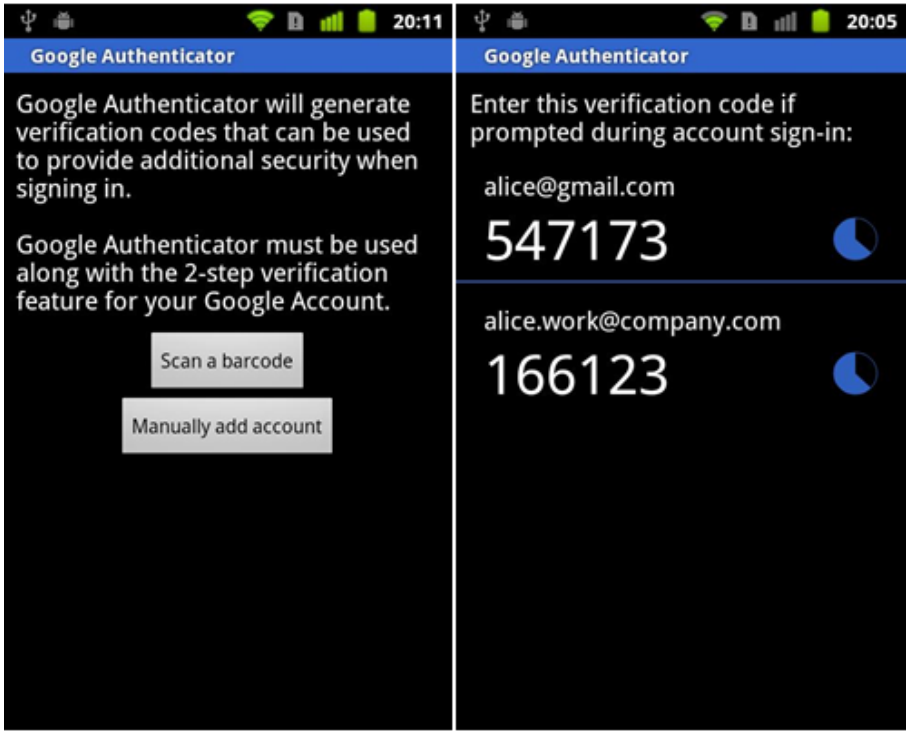
- The security of this scheme is based on wall clock time synchronization and an **initial shared secret** between Google and your phone.
- There is a **secure and practical way to share this secret** and at the same time configure the Google authenticator App.
  - Mobile Tagging
    - **QR codes**.

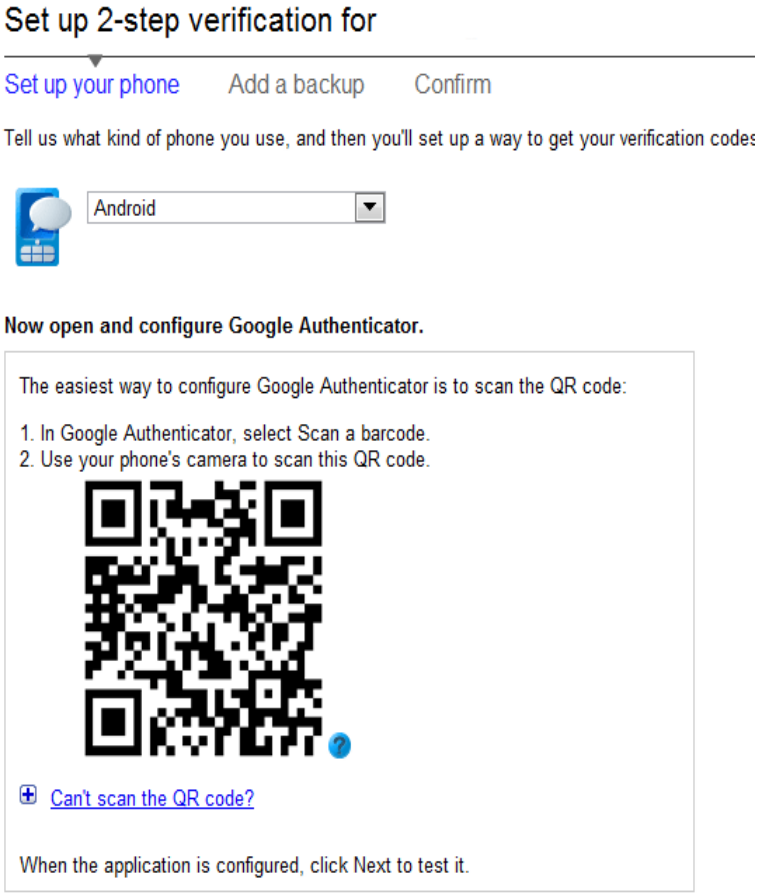
Mobile Tagging constitutes a very convenient mechanism to convey information into a mobile device

- Completely standardized as ISO/IEC 18004:2006
- Plenty of space for secret sharing and configuration purposes.
  - Numeric only Max. 7,089 characters
  - Alphanumeric Max. 4,296 characters
  - Binary (8 bits) Max. 2,953 bytes
- *A QR code is displayed on the screen and is then conveyed to the mobile phone through the mobile phone camera.*

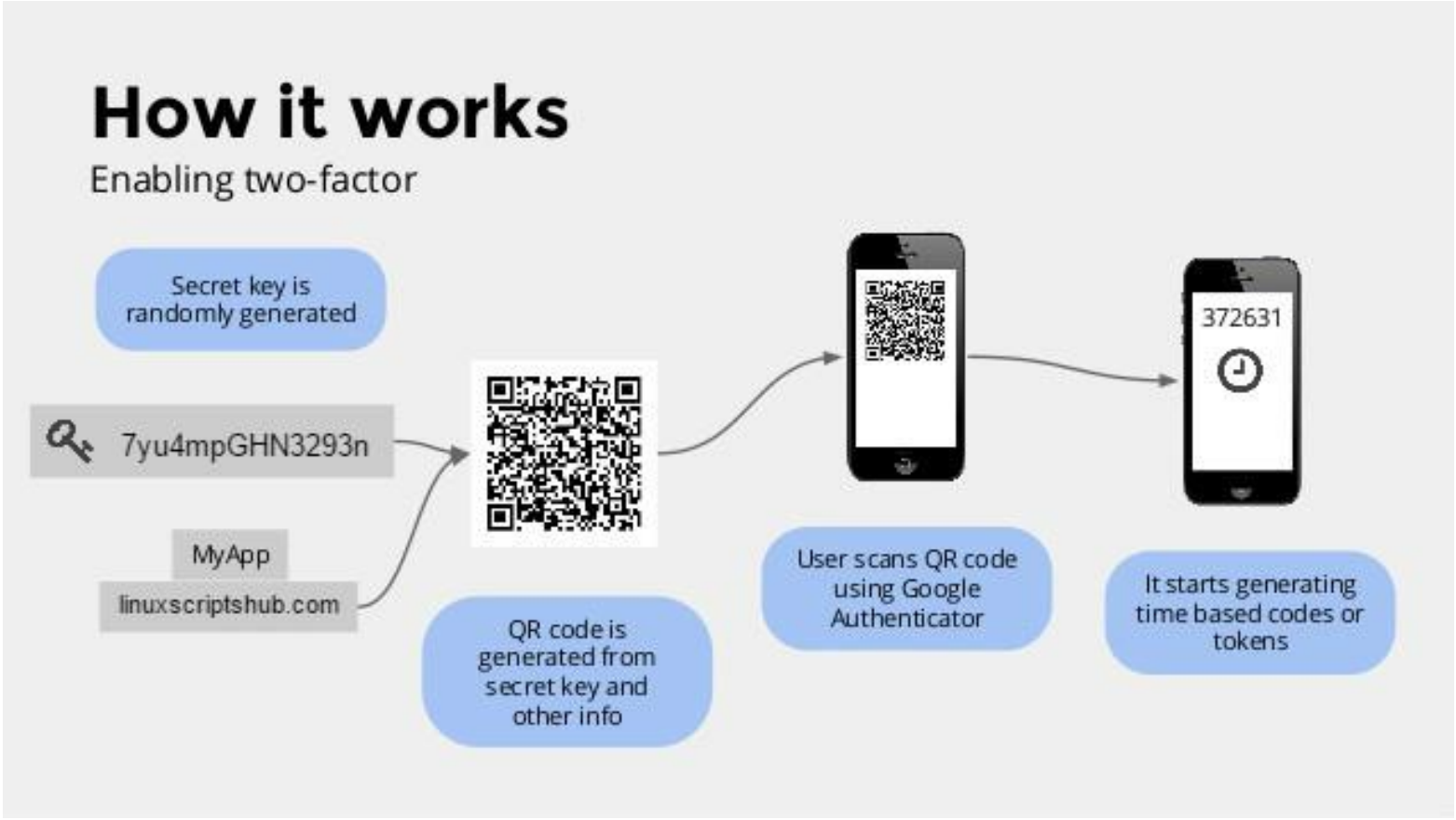
Mobile Tagging is a convenient mechanism to share secrets with a mobile device

- Google Authenticator and QR codes. A very versatile match.





Google Authenticator in a Nutshell



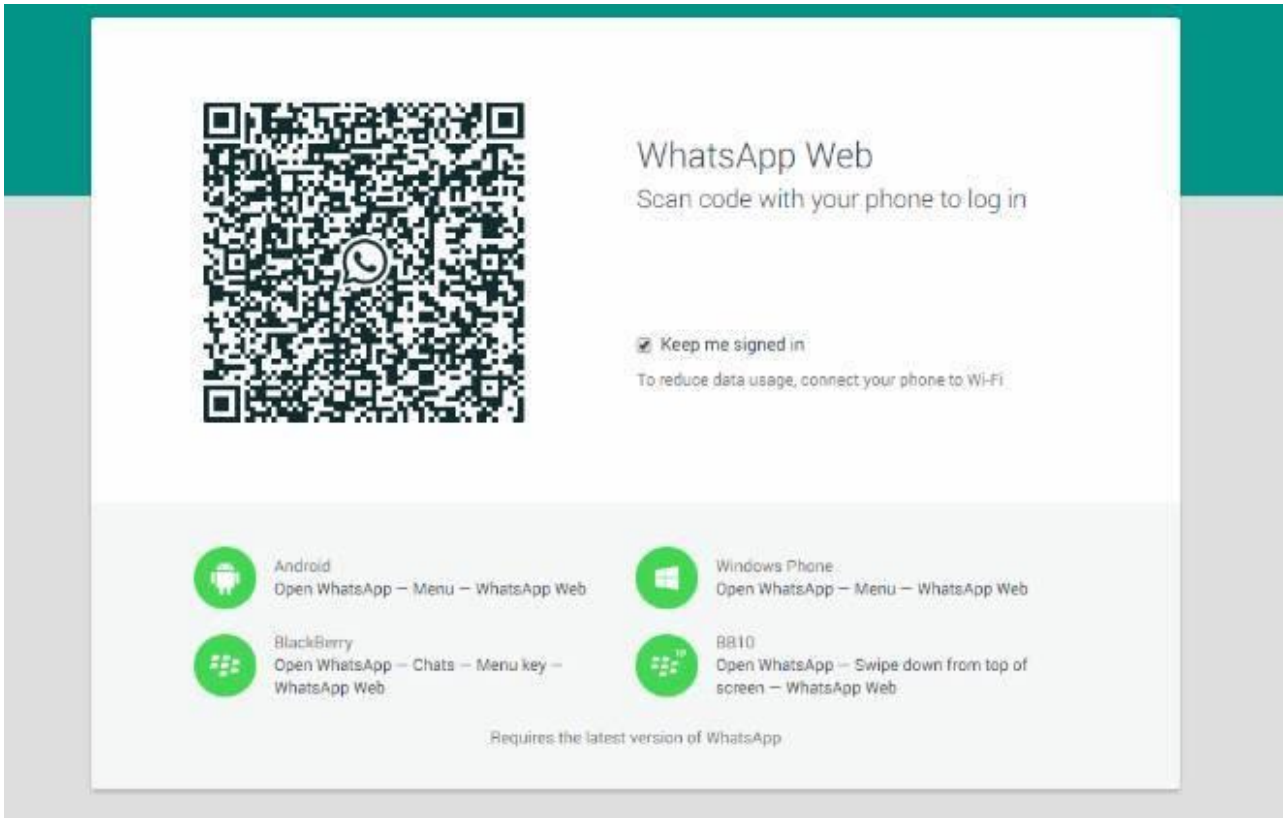
Google Authenticator OTP

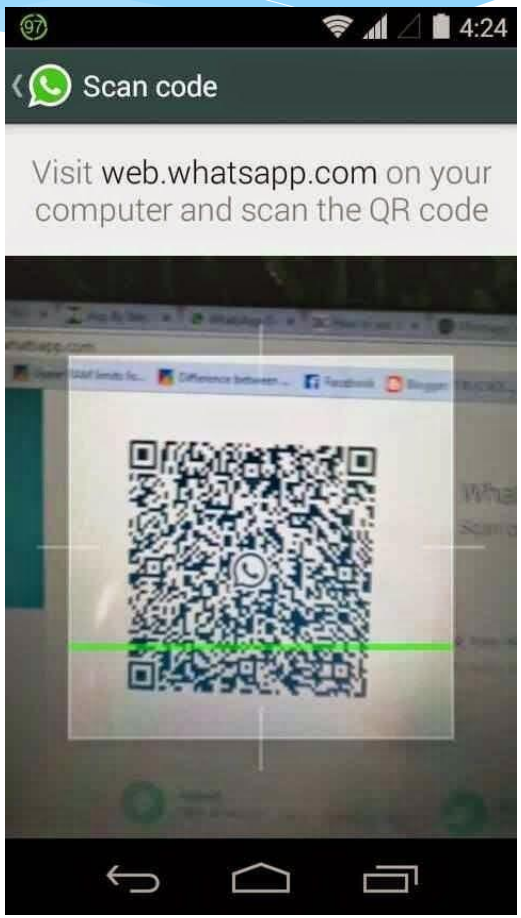
- The google Authenticator, once configured, does not need communication channels to provide the correct answer to the server challenge.
  - *Only correct wall time clock needed.*
- It solves the previously identified problems with SMS based schemes.
- Currently deployed at many high profile sites: *Gmail, Google Apps, DropBox, LastPass, Facebook, etc…*
- *The YubiTOTP Android Widget is able to generate an* ***OATH Timebased One Time Passcodes (TOTP)*** *from a secret stored in a YubiKey NEO (NFC enabled).*
- *What else can we do with this to improve security ?*
  - Mobile smart phones have Internet connectivity.

QR-Login/Authentication using an Internet Connected Mobile Phone

WhatsApp uses this idea to associate your smartphone to its web backend

https://web.whatsapp.com/





QR-Login main advantages

- With QR-codes the login process is quicker and more convenient than typing a username and password.
- Since the shared secret (the password) does not have to be memorized, or even typed in by a human, it can be long and complex.
- A virus-installed keylogger or shoulder-surfer cannot capture the password.
- The user can securely use an untrusted computer (such as one in an Internet cafe or hotel) without revealing their password.
- A phishing web site cannot capture the user password by tricking them into typing it in. The phone sends the shared secret, and will only send it to the web site in its database.
- By using different logins the user's account on one web site cannot be associated with the user's account on another web site.

QR-Login main advantages

- The password can be randomly generated.
  - If the user chooses to use a randomly generated username, the user's account on one web site cannot be associated with the user's account on another web site, again as happened in the Gawker password database spill.
- Users have more privacy options since it is easier to generate and recall random logins and passwords.
- The user will not lose access to a web site because they cannot remember a password.
- Since the authentication code is sent encrypted, and the web site authenticates itself to the user via HTTPS, the random secret can't be intercepted to authenticate another user's session.
- Finally, the login process is quicker and easier than typing a username and password.


fido™
alliance
simpler
stronger
authentication

The FIDO Alliance was formed in the summer of 2012

To Change Authentication Online by:

(a) Developing unencumbered Specifications that define interoperable mechanisms that supplant reliance on passwords

(b) Operating programs to help ensure industry adoption

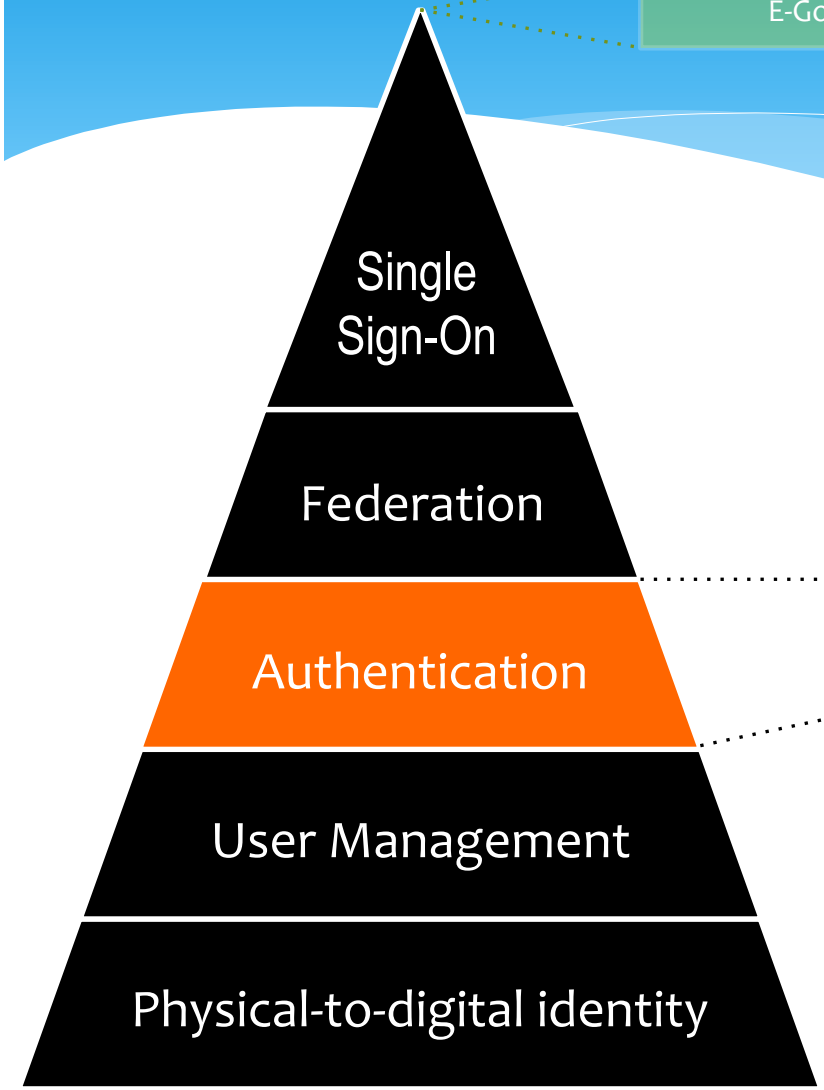
(c) Submitting mature Specifications for formal standardization

FIDO Alliance's Role…

* "Paper" Specifications
* Interoperability and Conformance testing
* Trademark licensing against criteria
* Thought leadership, nurture ecosystem
* The Alliance does not ship products!
* Implementations left to commercial vendors

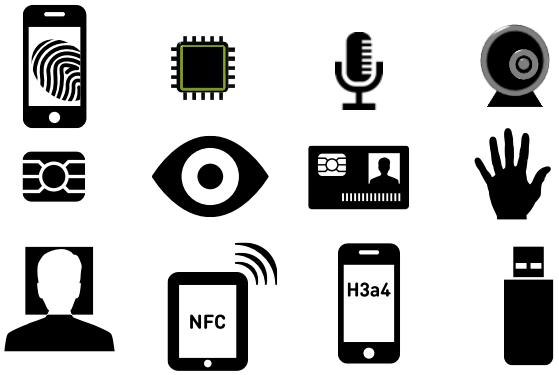
Identity & Authentication Building Blocks

E-Gov | Payments | Security | Personalization



MODERN AUTHENTICATION

Passwords | Strong | Risk-Based

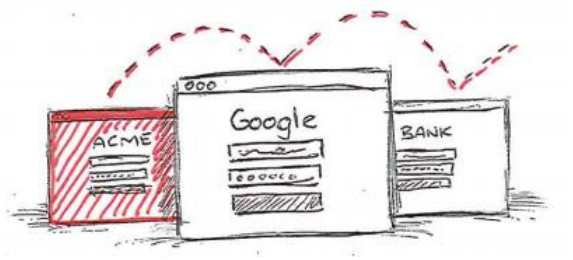


Why Authentication is Cybersecurity Priority #1

Poor authentication mechanisms are a commonly exploited vector of attack by adversaries; the 2013 Data Breach Investigations Report (conducted by Verizon in concert with the U.S. Department of Homeland Security) noted that 76% of 2012 network intrusions exploited weak or stolen credentials.

-- NIST Roadmap for Improving Critical Infrastructure Cybersecurity,12-Feb-2014

Today's Passwords



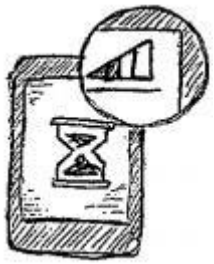
REUSED

PHISHED

KEYLOGGED

Today's Password Alternatives

One Time Codes with SMS or Device



SMS USABILITY

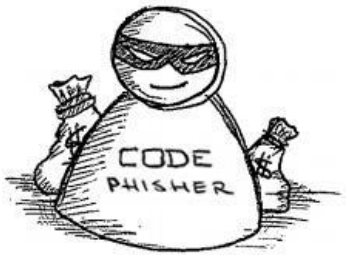
Coverage | Delay | Cost | Unsecure

DEVICE USABILITY

One per site | $$ | Fragile

USER EXPERIENCE

User find it hard



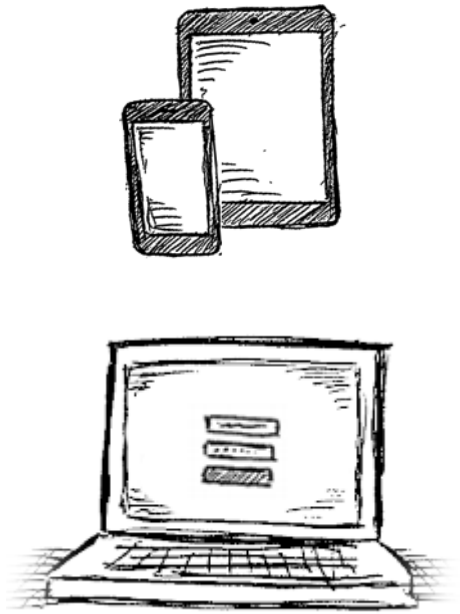
STILL PHISHABLE

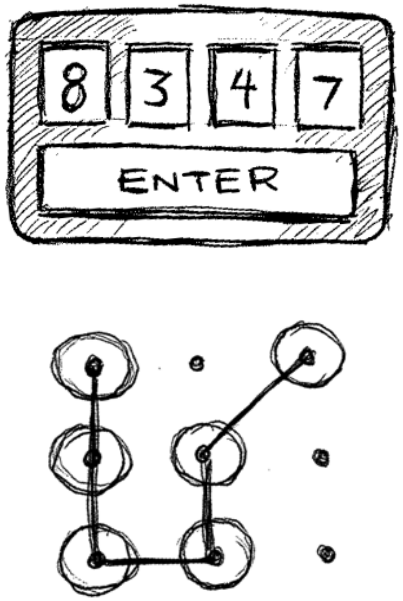
Known attacks today

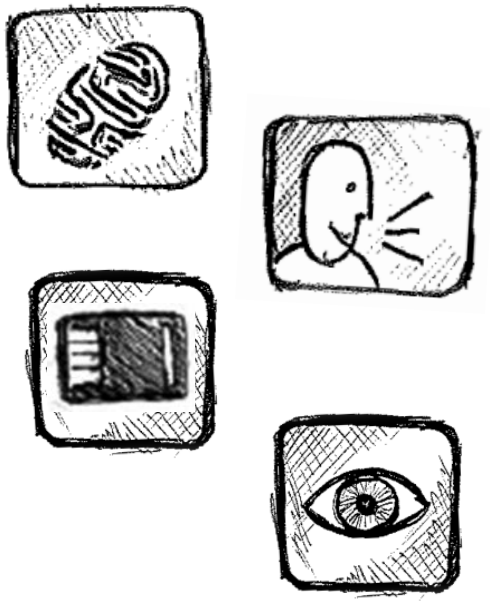
Major Industry Trend

Simpler, Stronger Local Device Auth

PERSONAL DEVICES	LOCAL LOCKING	NEW WAVE: CONVENIENT SECURITY
Carry Personal Data	Pins & Patterns today	Simpler, Stronger local authentication







Putting It all Together

The problem:

Simpler, Stronger online

The trend:

Simpler, Stronger local device auth

Why not:

Use local device auth for online Auth?

This is the core idea behind FIDO standards!