User Authentication

Authentication vs Identification

- Authentication is a binding between an *identity* and a *subject*, namely it is the process of verifying the identity declared by a subject:
  - ✓ *Identification step*: Presenting an identifier to the security system;
  - ✓ *Verification step*: Presenting or generating authentication information that corroborates the binding between the entity and the identifier (for example password)
- Thus we can say that
  - ✓ Identification is the means by which a user provides a claimed identity to the system
  - ✓ Authentication is the means of establishing the validity of the claim
  - ✓ If no one is able to obtain or guess the password of a user, then the combination user-ID and password enables administrators to set up the user permissions and audit his activities

Means of authentication

There are four general means for authentication

* Something that the individual **knows** (passwords, answers to predefined set of questions, PIN)
* Something that the individual **possesses** (badge, smart card, physical keys)
* Something the individual **is** (static biometrics) (fingerprint, retina, face)
* Something the individual **does** (dynamic biometrics) (voice pattern, handwriting characteristics, typing rhythm)

Authentication

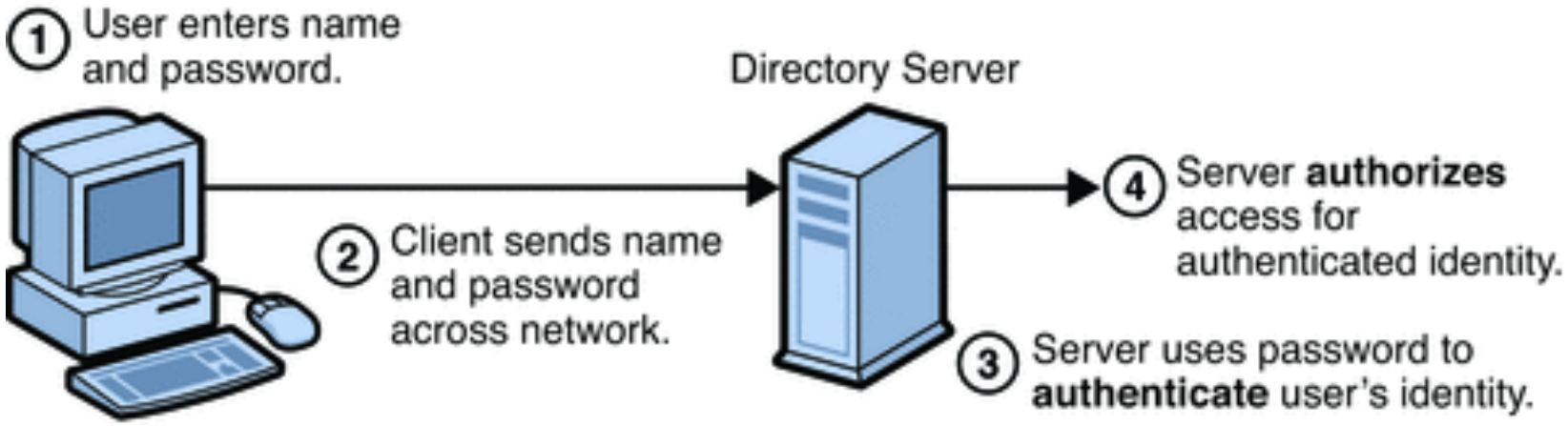
Authentication is important for

* Accountability (keep trace of the actions of the user -logs)
* Access Control (given the identity of the user we can establish what he/she can do)

Password Authentication

Widely used authentication method

* User provides username and password
* System compares password with that in password file



What is a password

A password is information associated with an entity that confirms the entity identity, it is a key-world known only to the entity and the system

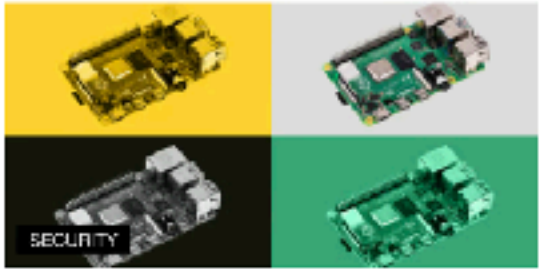
- The simplest password is a sequence of characters. The password space is the set of all sequences of characters that can be password
- Password may be combined with complementary information (set of information that the system stores and uses to validate the authentication information)
- Protection by password seems secure. However, common human errors reduce the security of password based systems

The top 4 instances when a weak password led to a major hacking incident

Updated on: 04 May 2023

Chris Stokel-Walker, Contributor

Editor's choice



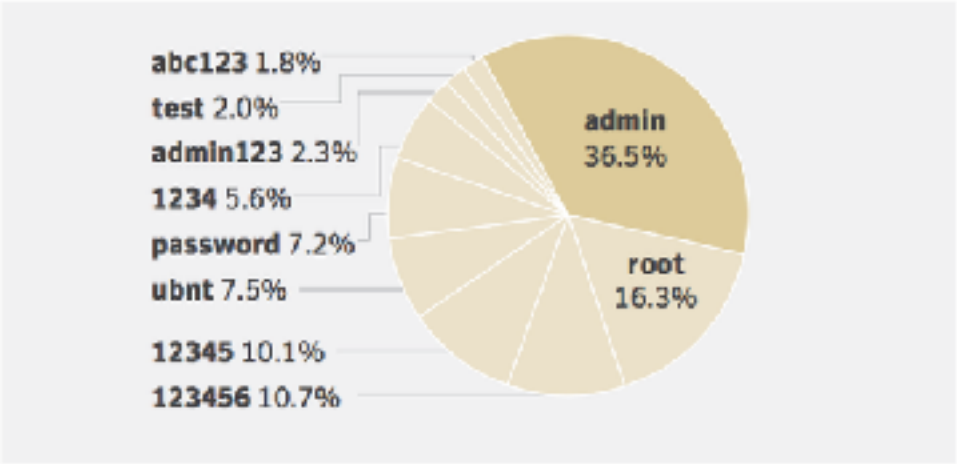
Experiment: anti-Pegasus box to keep spies away from my home

by Emmetas Napryo 06 October 2023

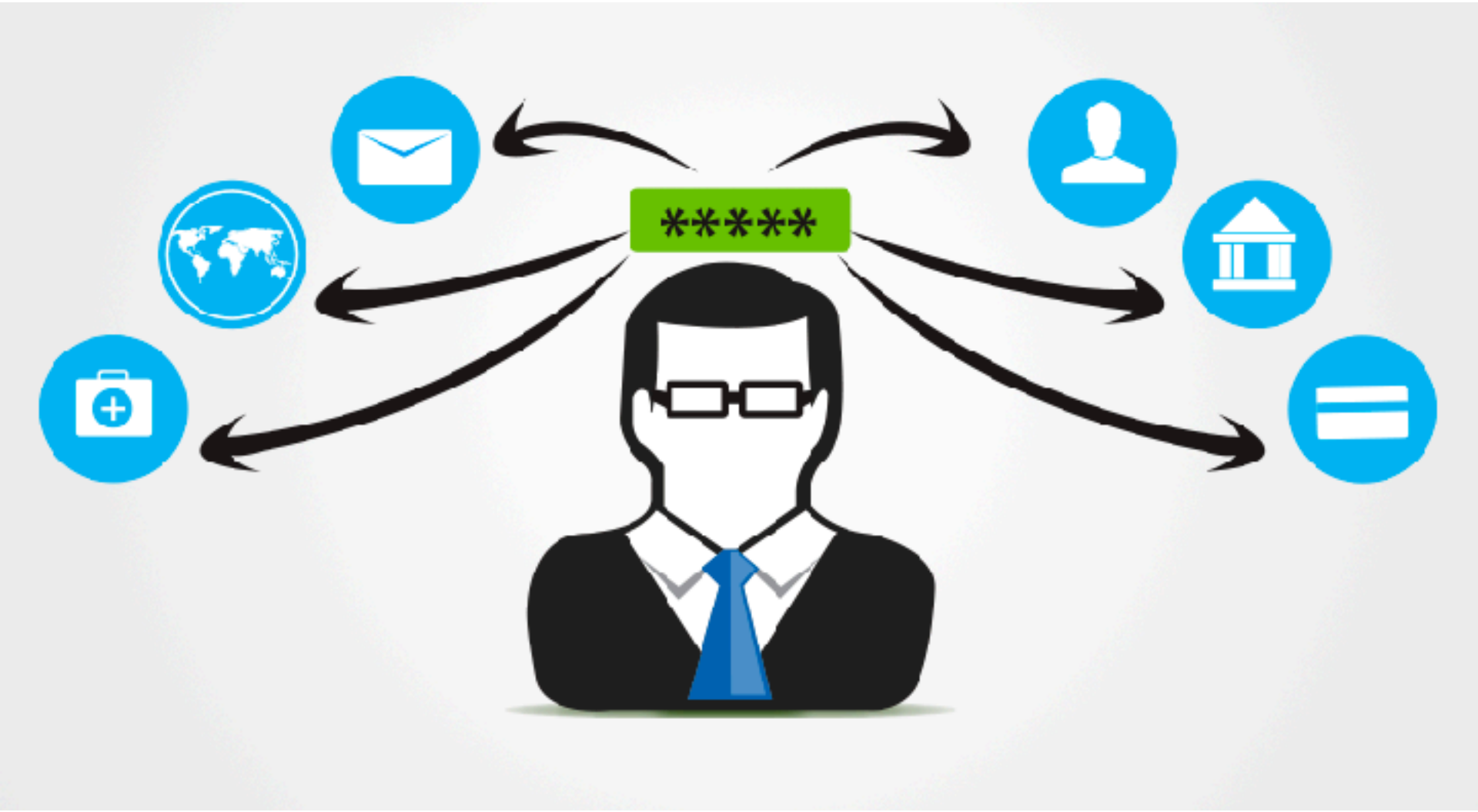
Journalists, activists, or minorities around the globe who are targeted by governments using high-tech spyware such as Pegasus have limited means to protect themselves. After recent revelations that ad

Mirai Botnet

- Mirai works by continuously scanning for IoT devices that are accessible over the internet
- It looks for devices that are protected by factory default or hardcoded user names and passwords
- It then infects them with malware that forces them to report to a central control server, turning them into a bot that can be used in DDOS attacks



Password Reuse



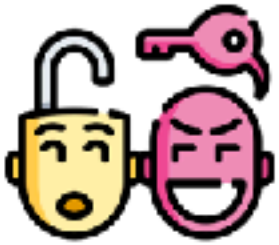
Predictable Password

Comics about work. Made with love & lots of coffee.
Now also on Webtoon & Tumblr. Or join r/workchronicles

Work Chronicles
workchronicles.com

How password are cracked?

Phishing



Social Engineering

Dictionary Attack

Rainbow Tables

Brute Force

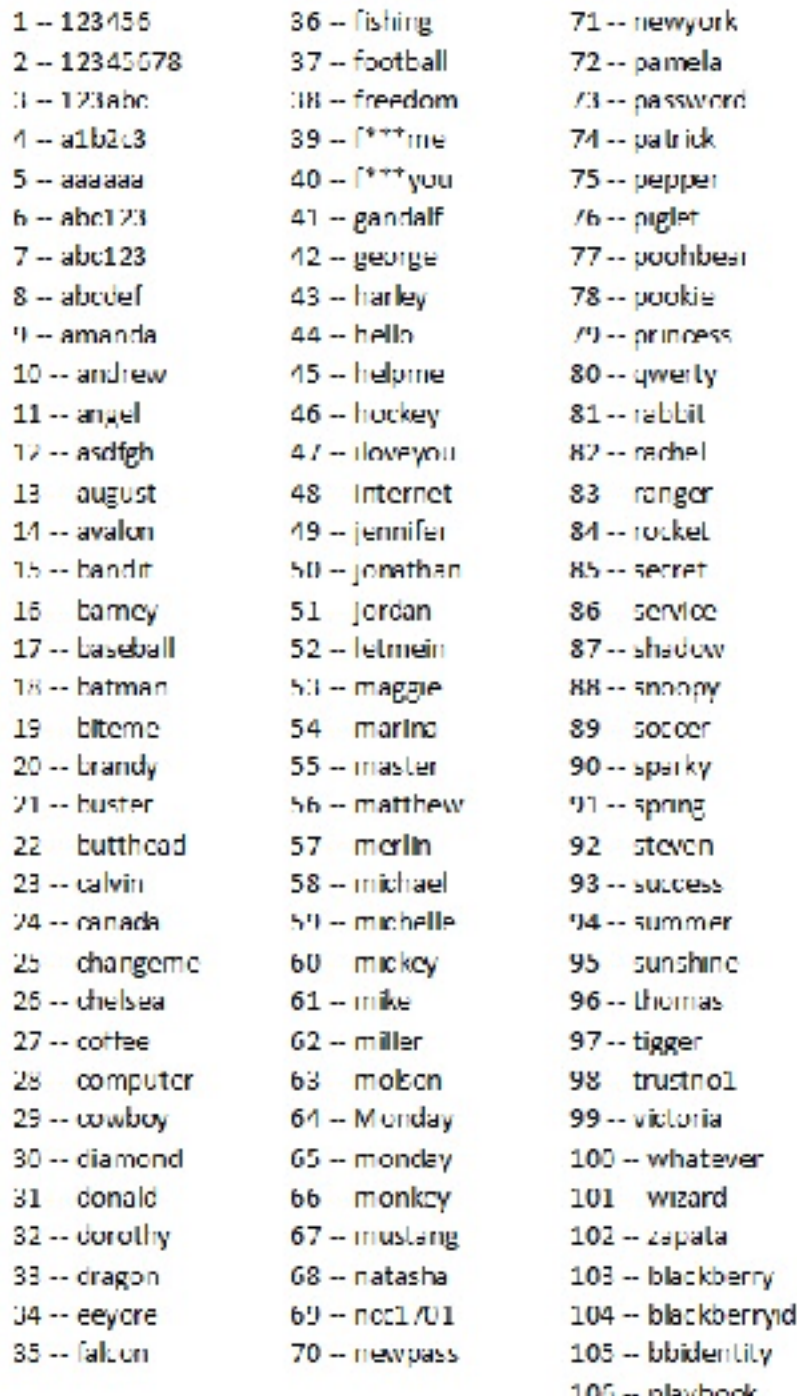
How are the pwd stored?

Brute Force

- Exhaustive search
  - ✓ The size of the password spaces is $|A|^n$
- Assume a 8 character password, with upper and lower case letters, digits, common symbols (96 possible characters)
  - ✓ 96^8 = 7.2 quadrillion password combinations
- Offline attack, the attacker has access to the encrypted material or a password hash and tries different key without the risk of discovery or interference.
- Online attack, the attacker needs to interact with a target system. In such cases, the system can counteract the attack by, for example, limiting the number of attempts that a password can be tried,

Dictionary attack

- Intelligent search
  - ✓ Try passwords associated with the user: name, name of friends, car brand
  - ✓ Try words in a dictionary
  - ✓ Try popular passwords
- Save attacker's time
- No guarantee the right password is found



Likely Password

- Password are often short, contain common words, easy to spell and pronounce! attackers exploit these features!
  - ✓ Attackers that use a brute force attack do it starting from shorter password and then considering longer ones
  - ✓ Often user select passwords of 3 or 4 characters
    - Try longer passwords, of length 5, requires only 475 seconds more
    - Password like *vxlag* or *msms* are less common than *entry* or *wine*
  - ✓ Users often use words that have a meaning
    - Most common words are around 80.000 and 80 seconds are enough for trying all of them

Common passwords

1. 123456, 123, 123123, 01234, 2468, 987654, etc
2. 123abc, abc123, 246abc
3. First Name
4. Favorite Band
5. Favorite Song
6. first letter of given name then surname
7. qwerty, asdf, and other keyboard rolls
8. Favorite cartoon or movie character
9. Favorite sport, or sports star
10. Country of origin
11. City of origin
12. All numbers
13. Some word in the dictionary
14. Combining 2 dictionary words
15. any of the above spelled backwards
16. aaa, eee, llll, 999999, and other repeat combinations

Likely passwords for the user

- Users often choose password that contain personal information, like the name of a family member, or of an object or animal that are familiar and therefore easy to remember

- If we reduce the set of possible guesses to names, street names, animals and so on we reduce the vocabulary that we need to try.
  - ✓ The attacker has an higher chance to succeed

- Sometimes users think to use a simple password and then replace o with 0, i with 1, e with 3, a with @ and so on…. This does not really provide higher defense since also the attackers know this trick

- Today most of the passwords used are weak!

Common Extensions

* Some sites force you to have passwords with both numbers and letters. For example Bob's password is *football*, and the site asks him to add some numbers to it to make it valid. Here's what people usually add.

    - Their year of birth / marriage / graduation (or expected grad) from HS or college
    - 007
    - 0 - 9
    - 69
    - 000, 111, 4444 or other long combinations
    - 123456, 123, 123123, 01234 and other retarded combinations

* Years are usually added in different ways: football85, football1985, football04 instead of football4. There's also the possibility of sub-connections like football_04 and football-84.

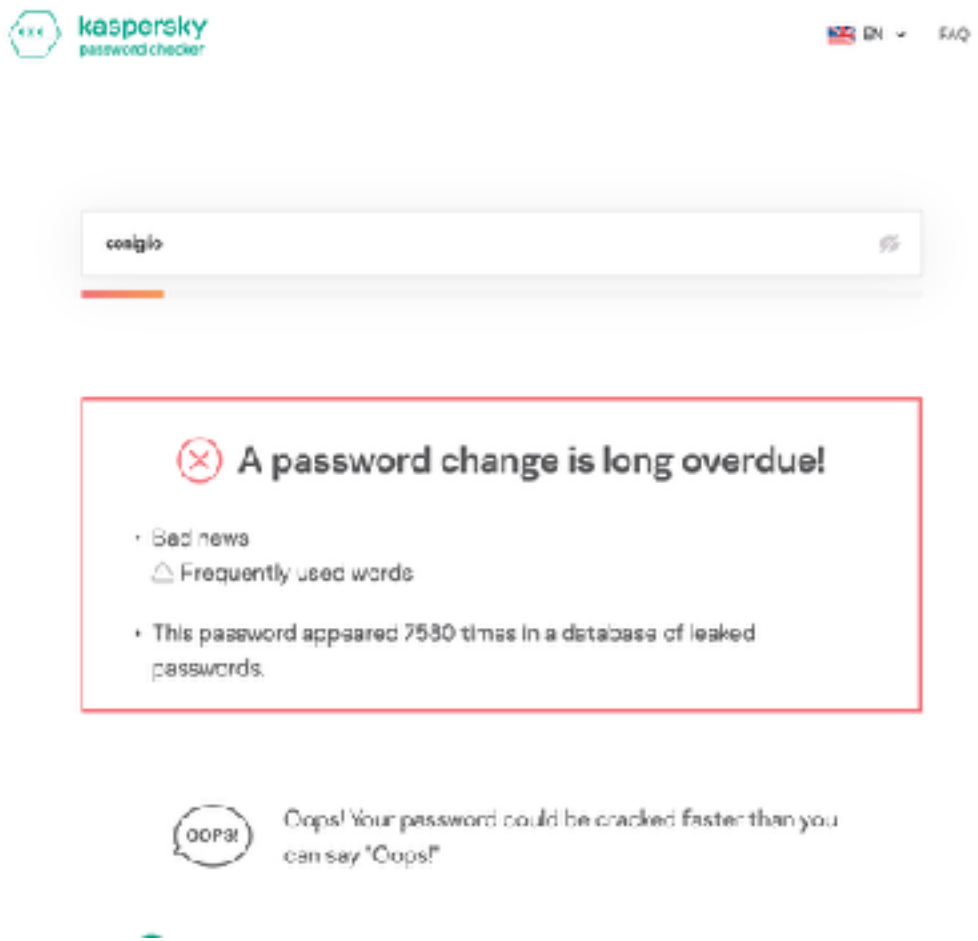
Rainbow Tables

- Precomputed compilation of plaintext passwords and matching hashes
- They are faster in cracking passwords than brute-force, dictionary attacks and hybrid attacks
- They occupy a lot of memory

John the Ripper

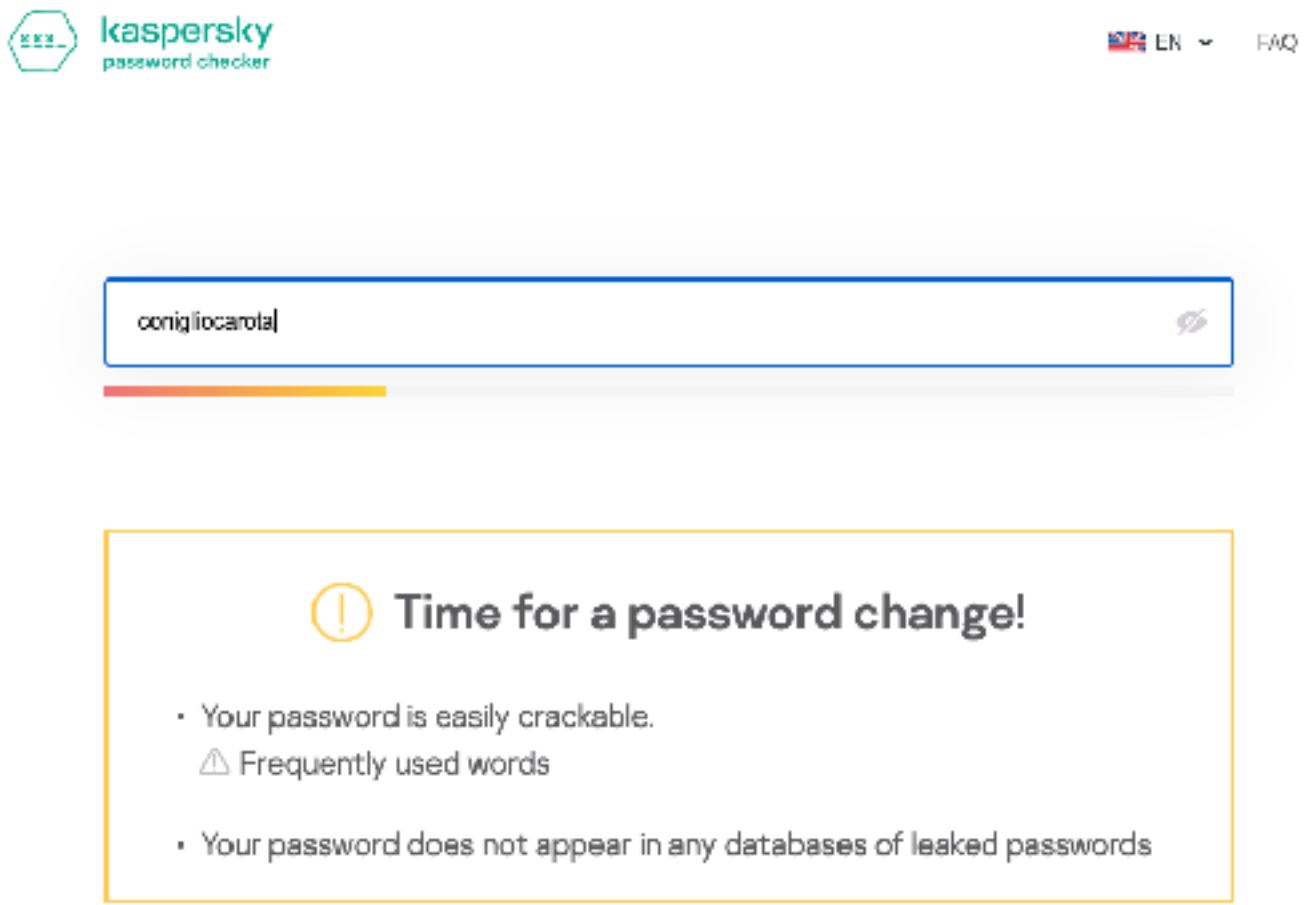
- Most popular password cracking tool
- Supports several common encryption technologies out-of-the-box for UNIX and Windows-based systems
  - UNIX crypt(3)
  - Traditional DES-based
  - Kerberos/AFS
  - Windows LM (DES-based)
  - DES-based tripcodes
  - SHA-crypt hashes (newer versions of Fedora and Ubuntu)
  - SHA-crypt and SUNMD5 hashes (Solaris)
- Three operation modes: singlecrack, wordlist, incremental

How long does it takes to crack a password?



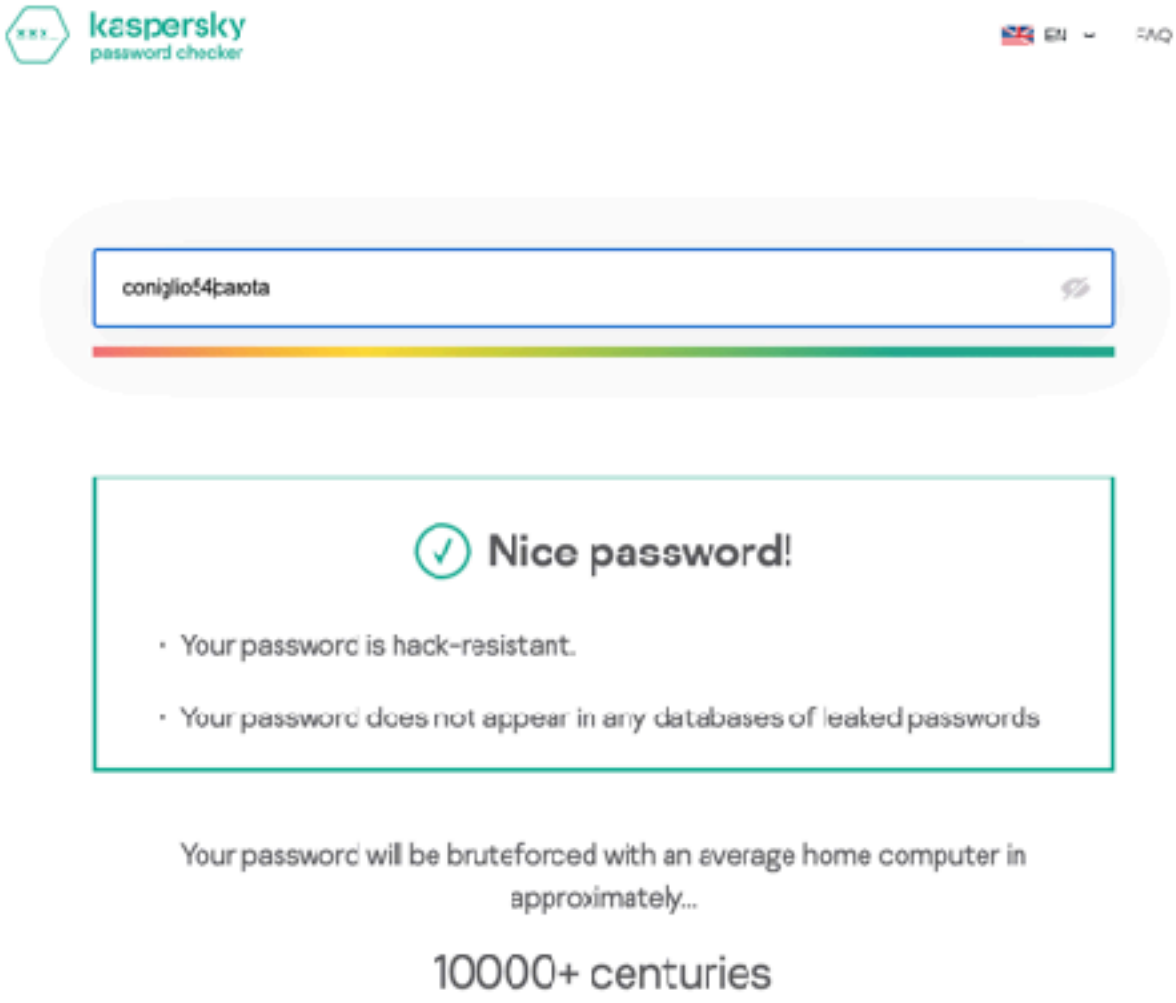
https://password.kaspersky.com/

How long does it takes to crack a password?



https://password.kaspersky.com/

How long does it takes to crack a password?

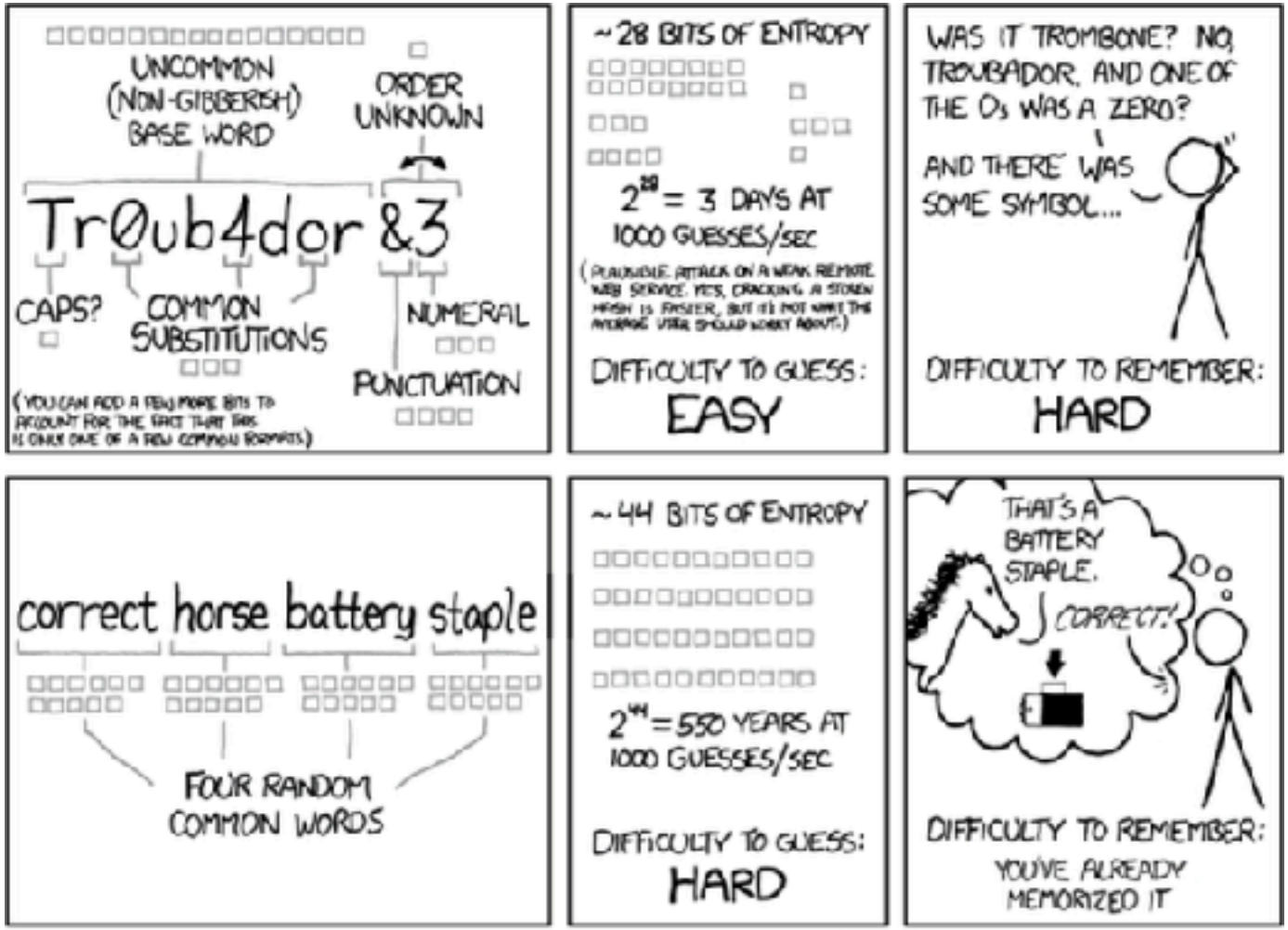


https://password.kaspersky.com/

Password Strength

- Password strength measure the effectiveness of a password against brute force attack
- It estimates the number of trials an attacker has to make to guess the password correctly
- It is normally computed as $|A|^n$
  - ✓ A is the set of symbols composing the password
  - ✓ n is the length of the password

Password Strength



THROUGH 20 YEARS OF EFFORT, WE'VE SUCCESSFULLY TRAINED EVERYONE TO USE PASSWORDS THAT ARE HARD FOR HUMANS TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.

TIME IT TAKES A HACKER TO BRUTE FORCE YOUR PASSWORD IN 2023

Number of Characters	Numbers Only	Lowercase Letters	Upper and Lowercase Letters	Numbers, Upper and Lowercase Letters	Numbers, Upper and Lowercase Letters, Symbols
4	Instantly	Instantly	Instantly	Instantly	Instantly
5	Instantly	Instantly	Instantly	Instantly	Instantly
6	Instantly	Instantly	Instantly	Instantly	Instantly
7	Instantly	Instantly	1 sec	2 secs	4 secs
8	Instantly	Instantly	28 secs	2 mins	5 mins
9	Instantly	3 secs	24 mins	2 hours	6 hours
10	Instantly	1 min	21 hours	5 days	2 weeks
11	Instantly	32 mins	1 month	10 months	3 years
12	1 sec	14 hours	6 years	53 years	226 years
13	5 secs	2 weeks	332 years	3k years	15k years
14	52 secs	1 year	17k years	202k years	1m years
15	9 mins	27 years	898k years	12m years	77m years
16	1 hour	713 years	46m years	779m years	5bn years
17	14 hours	18k years	2bn years	48bn years	380bn years
18	6 days	481k years	126bn years	2tn years	26tn years

HIVE SYSTEMS

> Learn how we made this table at hivesystems.io/password

Online Dictionary Attack

- Password Intelligent search
  - Try passwords associated with the user e.g name, name of friends, car brand
  - Try words in a dictionary
  - Try popular passwords
- Save attacker's time
  - No guarantee the right password is found

Possible Countermeasures

- Password policies
  - ✓Set password length: minimal password length should be prescribed
  - ✓Set password format: mix upper and lower case symbols, numerical, and non-alphabetical symbols
  - ✓Avoid obvious passwords: 12345, Forever1, Jhon3:16, Monster1…
- Changing password:
  - ✓Force users to change passwords regularly (30,60,90 days)
- Machine generated passwords
  - ✓Pronunciabile passwords are generated for the user

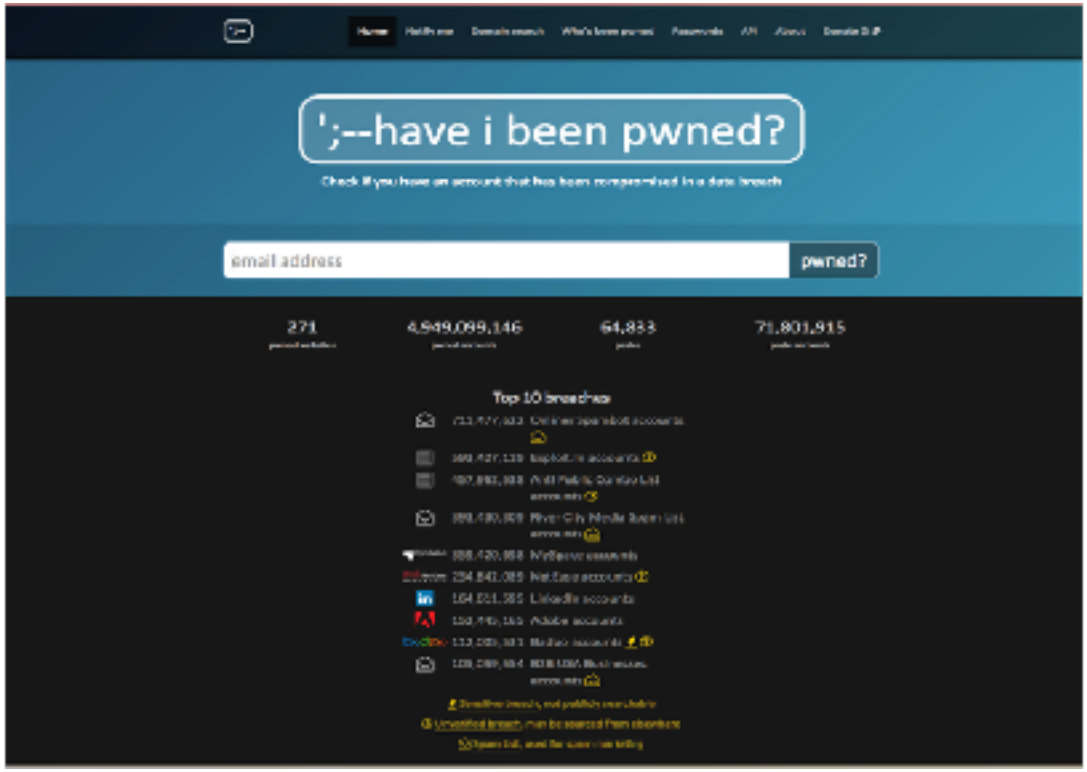
Possible Countermeasures

- Password ...ies
  - ✓Set ... length: minimal ... length should ...eed
  - ✓Set pass... : mix ... lower case symbols, nu... ...etical symbols
  - ✓Avoid obviou... ...2345, Forever1, Jhon3:16, Monste...
- Changing passwo...
  - ✓Force use... ...rds regularly (30,60,9...
- Machine ...asswords
  - ✓Pront... passwords are g... for the user

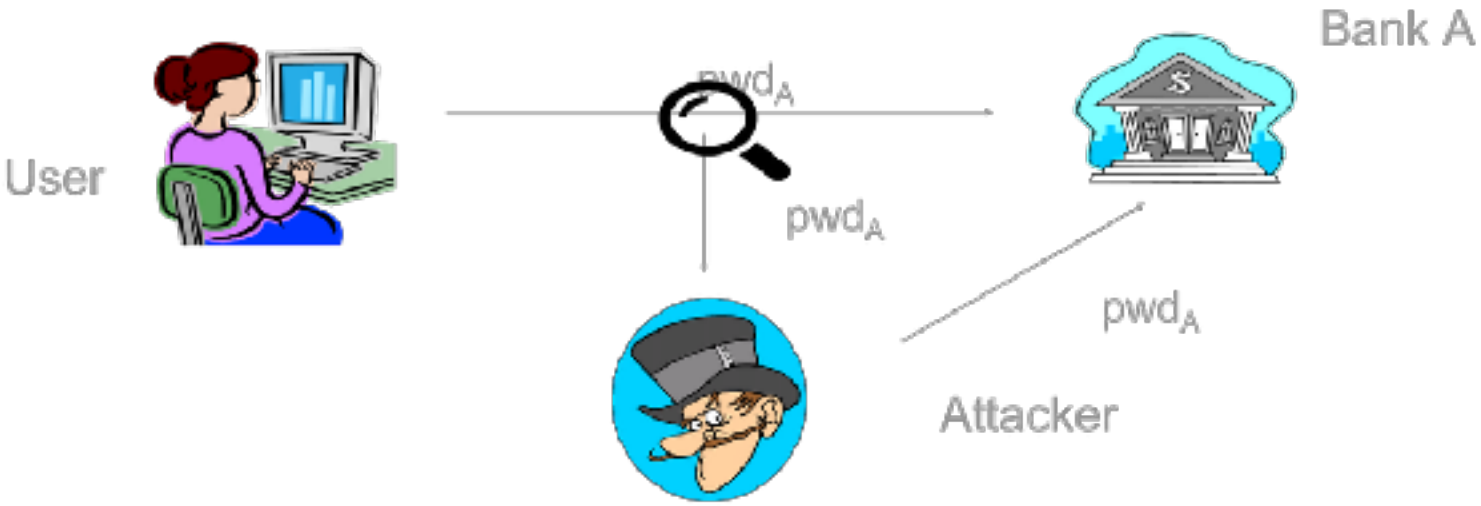
Possible Countermeasures

- Lockout mechanics
  - ✓ Lock user account after several unsuccessful login attempts
- Throttling
  - ✓ Time delays are introduced between consecutive failed login attempts
- Protective monitoring
  - ✓ Monitoring login to detect unusual use
  - ✓ Notify the user with details of attempted login
- Password blacklisting
  - ✓ Check if an input password is in a list of common words
- Multi-factor authentication

Password blacklisting



Interception



Keylogger

- Small program that monitors each keystroke the user types on his keyboard
- Installed by attaching the program to an image or file and then send it via email
- Popular keyloggers
  - ✓Refog
  - ✓Revealer
  - ✓KidLogger

“Leaking” users

The easiest way to have a password is to receive it directly form the user

- Users write their passwords on notes, share password with colleagues, use the same passwords multiple times
- There exists attacks that attempt to obtain password or sensitive information related to passwords directly form the users:
  - ✓ *Social engineering* involves the study of the behaviour of users in order to steal information form him/her

Social Engineering

- Phishing emails
  - ✓ Attackers send an email asking to reset password
- Shoulder-surfing
  - ✓ Attacker gathers passwords by watching over a person's shoulder while he/she is logging in
- Dumpster-diving
  - ✓ Attacker look into the trash for piece of papers or documents with written passwords
- Countermeasure: User Awareness and Training

Protecting Passwords

Password have a problem: they are reusable

* Password aging: tries to ensure that by the time the password is guesses it is no longer valid
* One-Time Password (OTP): a particular case of password aging are the one time password. These are passwords that are valid only for one login session or transition
* They are not vulnerable to replay attacks: an attacker that steals a password cannot use it since it is no longer valid

Password Aging

Password aging is the requirement that a password be changed after some period of time has passed or after some event has occurred

- Assume that the expected time to guess a password is 180 days. Then, changing the password more frequently than 180 days will, in theory, reduce the probability that an attacker can guess a password that is still being used
- In practice, aging by itself ensures a little, because the estimated time to guess a password is an average
- If users can choose passwords that are easy to guess, the estimation of the expected time must look for a minimum, not an average.
- Password aging works best in conjunction with other mechanisms

Password Aging

- There are problems in implementing password aging:
  - ✓ Force users to change passwords
  - ✓ Provide notice of the need to change password and a user-friendly method for changing password
  - ✓ Password aging is useless if the user can change the password to the same thing
    - record the last **n** inserted passwords
    - problem: user can change password very quickly and then use the same password
    - an alternative approach is based on *time*: the user must change the password to one other than the current password. The password cannot be changed for a minimum period of time

One Time Password (OTP)

One time password is a password that is invalid as soon as it is used

- It is not vulnerable to replay attacks, namely if an attacker obtains a password it cannot be used again
- It needs a token (something that the user has) to generate a random one time password
  - ✓ Use mathematical algorithms that build new password from previous ones
- The user and the systems have to be synchronized
- One time passwords are distributed through HW support (token), SMS, written on paper…

Mechanisms

- Assume that the users has chosen a strong password and that he/she stores it in a secure way….
- … the systems has to protect the password during the authentication process:
  - password insertion
  - store of the password in the system

Password Insertion

- Project secure interfaces: the *insertion field* of password should be protected
- Do not *send* passwords in clear text for the authentication on the server
- Protect the *machine* where the login occurs: the machine could be compromised with a malware that tries to steal the pair **<userID, password>**
- *Trusted path:* facility that ensures that the users is "speaking" with the trusted system

Password Insertion

- The response time are constant: the systems check the password one character at the time and at the first error they reject the password
- This can be exploited by the attacker: by observing the response time of the system it is possible to understand how many characters were correct
- Longer passwords require longer time of research

System protection

- Protect the memory area where the local copies of the passwords are memorized
- Protect the log file of the wrong passwords typically used in the auditing process
  - ✓ If the log file is not protected properly the attack becomes easy
  - ✓ identify a record for which the access has been denied for wrong user-id
  - ✓ there is an high probability that the password is correct and that the real user-id is a variant of the wrong one (a typo made by the user)

Protect the local copies of pwd

In order to verify a password the system has to be able to compare the inserted passwords with the real passwords

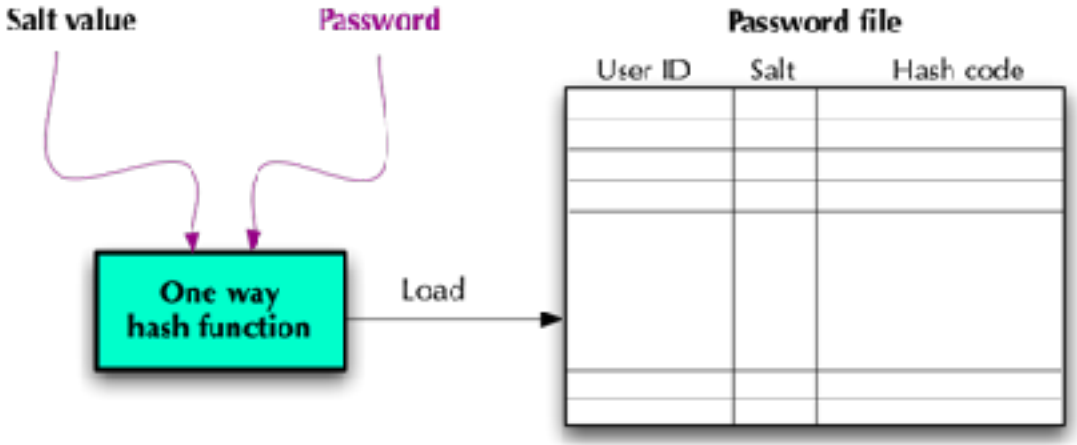
- The attacker can try to access the file containing the passwords of the system
  - ✓ In some systems this file contains a table with two column relating user-id and passwords
- It is important to protect this file
  - ✓ Restricted access control, only the OS can access the file. Not all the modules of the OS need to access the file of passwords
  - ✓ Allow to access the table only the modules that need to do it
  - ✓ Backup can be exploited by attackers

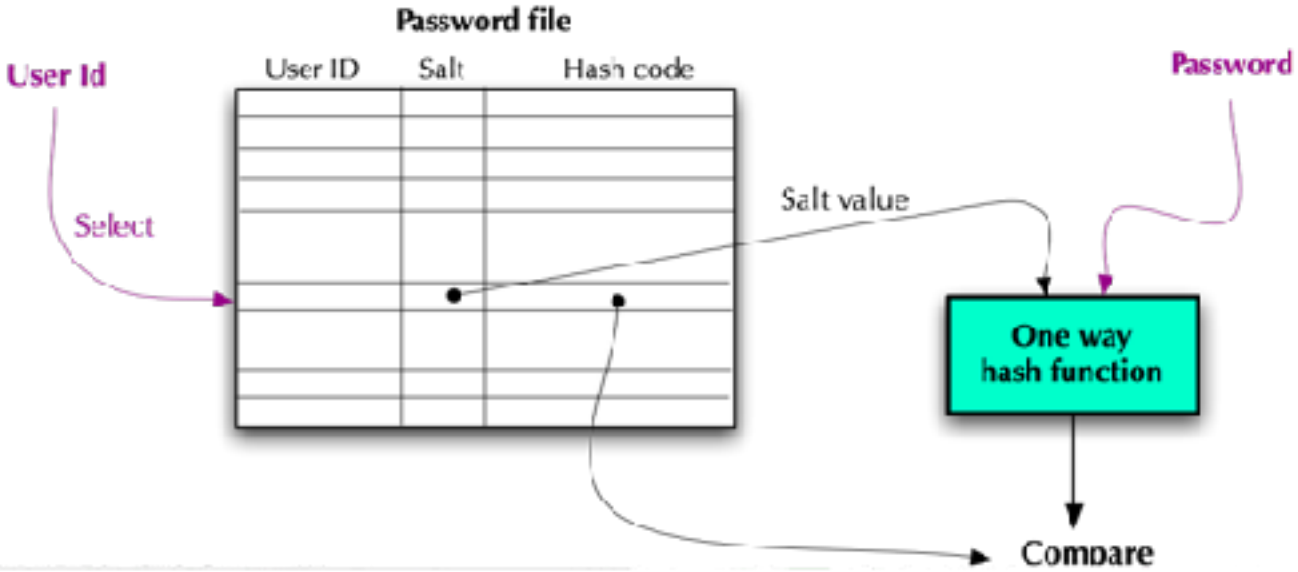
Crypted password list

- In order to protect the list of passwords from intruders we can crypt the list
- Standard cryptography: all the tables is encoded or only the column of the password.
  - When the user inserts the password the system decodes the crypted password and then compares them
  - The password is present in clear in memory for a short amount of time
- One-way cryptography: The passwords in the table are crypted and stored.
  - When the user inserts a password the system crypts it and then compares it with the one in the table
  - Ensure that it is not possible for two passwords to have the same encoding

Salt value

password insertion





password verification

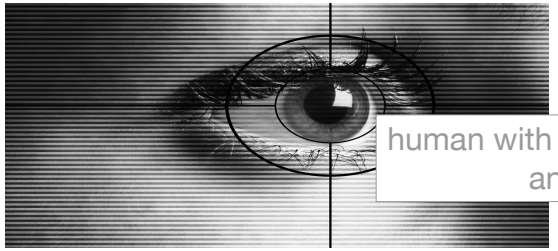
Summary

- Password based authentication systems are not secure
  - ✓Users use ease to guess passwords
  - ✓Users reuse passwords across multiple web sites
- Password based authentication systems are vulnerable to various attacks
- Social engineering and data breaches are on top of the list
- Effective countermeasures are
  - ✓Account lockout and throttling
  - ✓Predictive monitoring
  - ✓Password blacklisting
  - ✓Multi-factor authentication

Other means of authentication

Authentication

- The determination of identity, usually based on a combination of
  - ✓ something the person knows (like a password),
  - ✓ something the person has (like a smart card or a radio key for storing secret keys),
  - ✓ something the person is (like a human with a fingerprint)



human with fingers and eyes

Something you are

password=uclb()w1V
mother=Jones
pet=Caesar

Something you know

radio token with secret keys

Something you have

Token based authentication

- Objects that the user possess for the purpose of user authentication are called tokens
- Tokes are used to electronically prove the identity of user (for example to access the bank account). It is often used combined with password.
- Various types of tokens
  - ✓ Barcodes
  - ✓ Memory cards / magnetic stripe cards
  - ✓ Smart cards
  - ✓ One time password (OTP) devices

Barcode

- Developed in the 20th century to improve efficiency in grocery checkout.
- First-generation barcodes represent data as a series of **variable-width, vertical lines** of ink, which is essentially a one-dimensional encoding scheme.
- Some more recent barcodes are rendered as **two-dimensional patterns** using dots, squares, or other symbols that can be read by specialized optical scanners, which translate a specific type of barcode into its encoded information

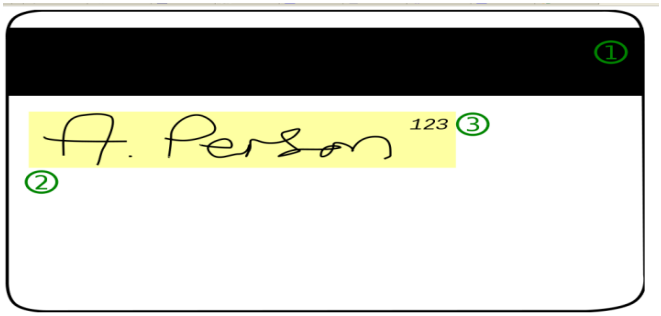
Authentication via Barcode

- Since 2005, the airline industry has been incorporating two-dimensional barcodes into boarding passes, which are created at flight check-in and scanned before boarding.
- In most cases, the barcode is encoded with an internal unique identifier that allows airport security to look up the corresponding passenger's record with that airline.
- Staff then verifies that the boarding pass was in fact purchased in that person's name (using the airline's database), and that the person can provide photo identification.
- In most other applications, however, barcodes provide convenience but not security. Since barcodes are simply images, they are extremely easy to duplicate.

Two-dimensional barcode

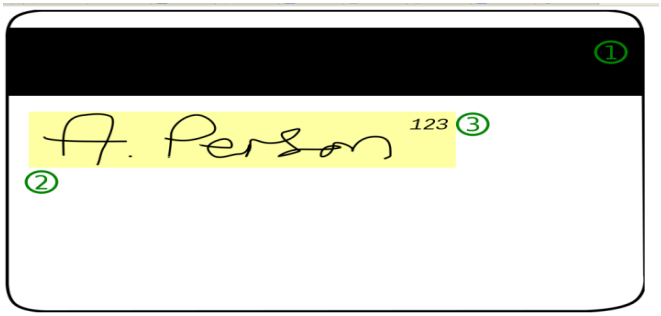
Memory card

- Plastic card with a magnetic stripe containing personalized information about the card holder.
- The most common such card is the *bank card* with a magnetic stripe in the back. A magnetic stripe can store only a *simple security code,* which can be read (and unfortunately reprogrammed) by an inexpensive card reader
- The first track of a magnetic stripe card contains the cardholder's full name in addition to an account number, format information, and other data.
- The second track may contain the account number, expiration date, information about the issuing bank, data specifying the exact format of the track, and other discretionary data
- Memory cards can store but not process data



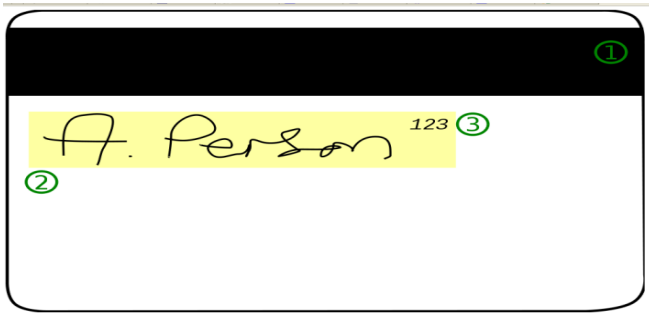
Memory card

- One vulnerability of the magnetic stripe medium is that it is easy to read and reproduce. Magnetic stripe readers can be purchased at relatively low cost, allowing attackers to read information off cards.
- When coupled with a magnetic stripe writer, which is only a little more expensive, an attacker can easily clone existing cards.
- So, many uses require card holders to enter a **PIN** to use their cards (e.g., as in ATM and debit cards in the U.S.).
- Memory cards can be *used alone for physical access* (ex: hotel room)
- The memory card, when combined with a PIN or password, provides significantly greater security than a password alone. *Indeed an attacker has to get physical access to the card*



Memory card

- The drawbacks of memory cards are the following:
  - Requires special reader: this increases the cost of using the token and creates the requirement to maintain the security of the reader's HW and SW
  - Token loss: A lost token temporarily prevents its owner from gaining system access. Moreover, there is an administrative cost in replacing the token. If an attacker finds the token he/she has only to guess the PIN
  - user dissatisfaction: Users may have no difficulty in accepting the use of a memory card for ATM access, its use for computer access may be deemed inconvenient



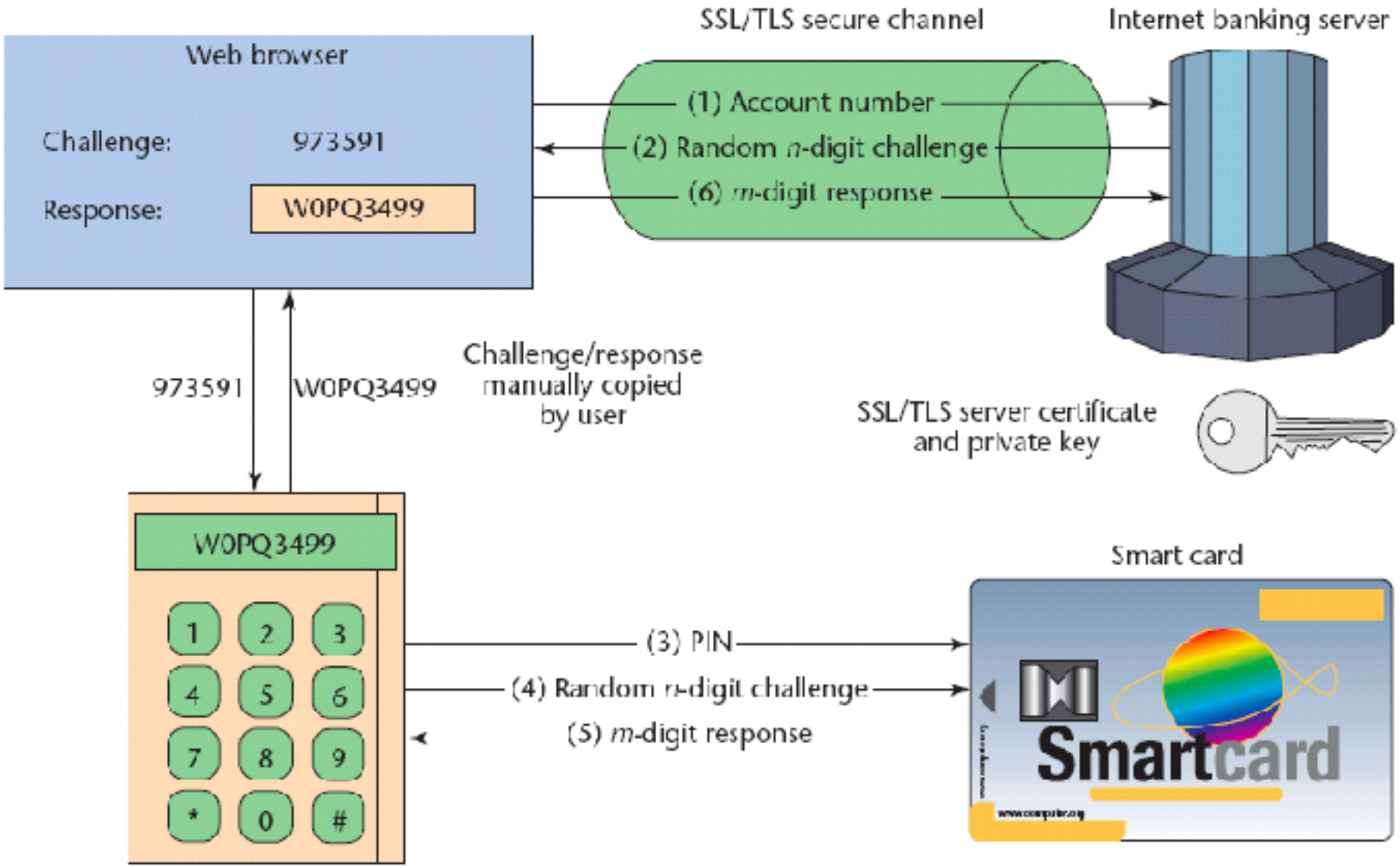
Authentication via OTP

- Single factor OTPs
  - Generated a number every 30 or 60 seconds
  - Embed a secret that is used to generate the OTP
- Multi factor OTPs
  - They use a second factor authentication
  - They also contain a symmetric key and a nonce
  - An encryption algorithm is applied to obtain the OTP

Smart Card

- Smart card has the appearance of a credit card, has an electronic interface, and may use any of the protocols describe earlier
- Smart card contain a *microprocessor* (including processor, memory, I/O ports).
- Challenge-response protocol with the reader
  1. Users enters a PIN
  2. Reader sends a challenge B
  3. Smart card generates random value A, A || B and sign (A||B) with the private key
  4. Reader verifies the signature with public key

Challenge Response

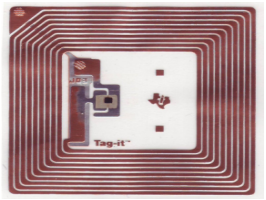


RFID tags

Authentication Server

Reader

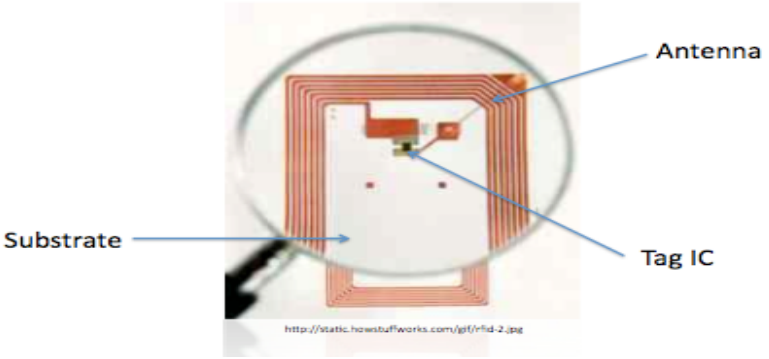
0100101110100...



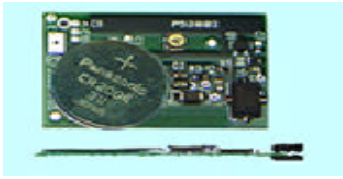
RFID Tag

Type of tags

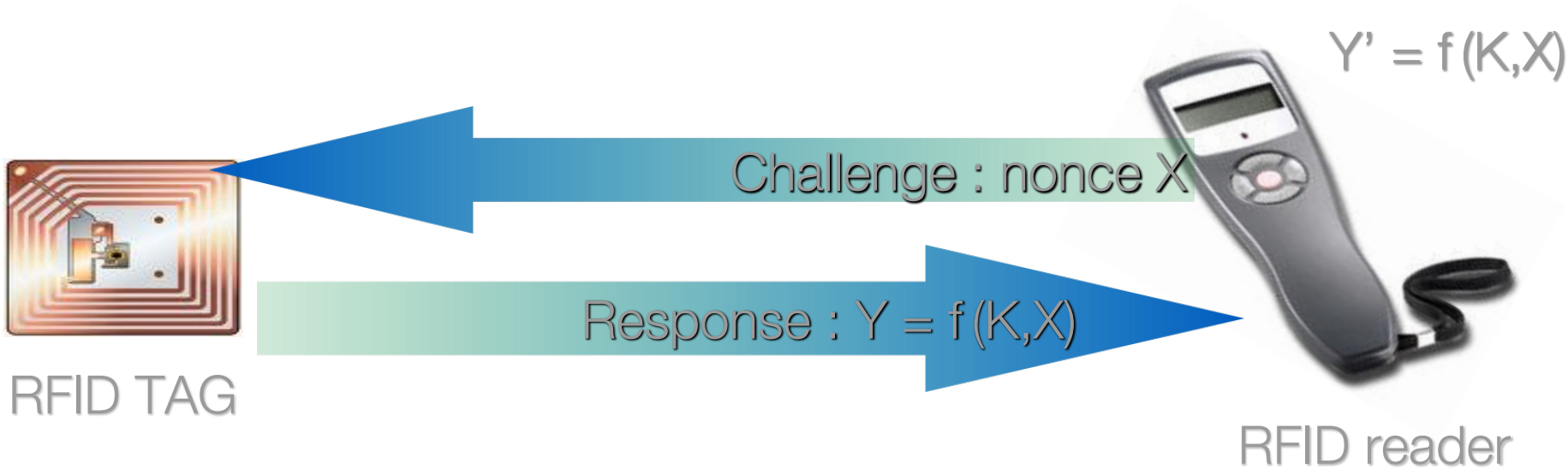
* **Passive:** Operational power scavenged from reader radiated power



* **Active:** Operational power provided by battery - transmitter built into tag

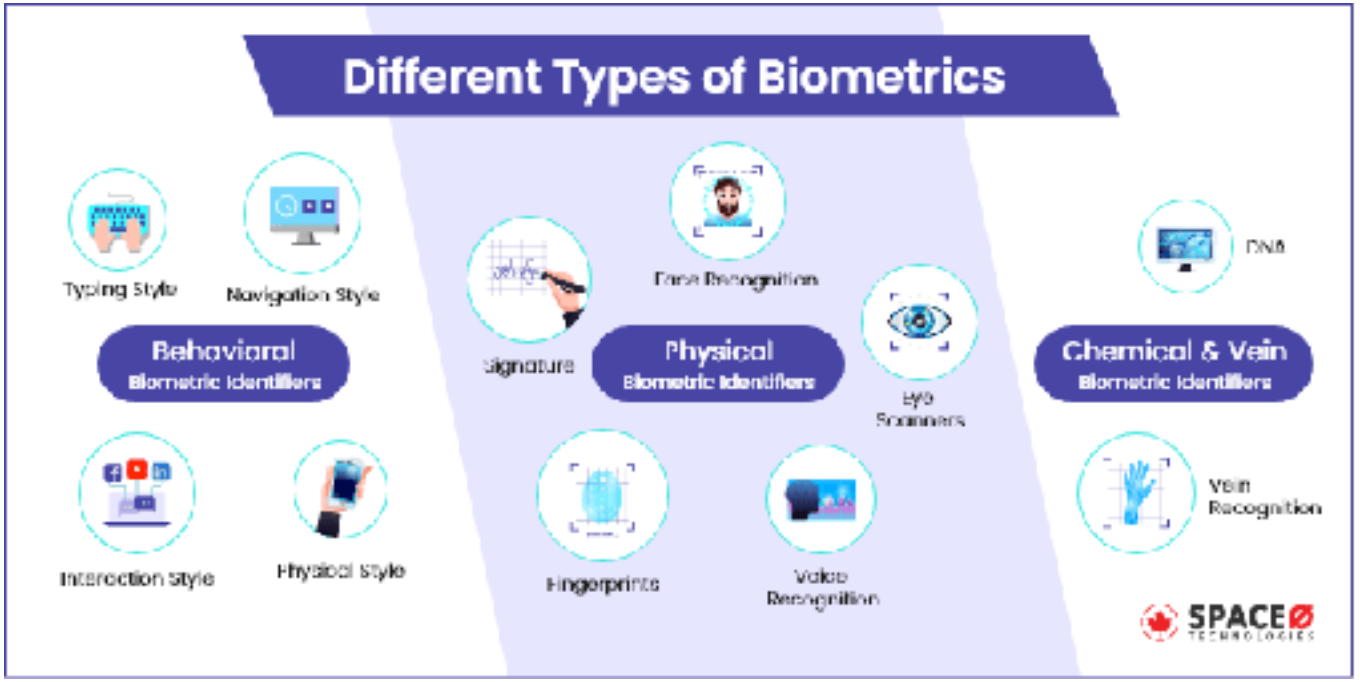


Authentication via RFID tags



- ✓ Function f is public
- ✓ Secret key K is known only to the tag and the reader
- ✓ The reader sends challenge X and the tag responds with Y
- ✓ The reader computes Y' = f(K,X) and verifies that Y=Y'

Biometric Authentication



Requirements for biometric authentication

- **Universality:** Almost every person should have this characteristic.
- **Distinctiveness:** Each person should have noticeable differences in the characteristic.
- **Permanence:** The characteristic should not change significantly over time.
- **Collectability:** The characteristic should have the ability to be effectively determined and quantified.

Operations of a biometric authentication

- Enrollment: Each individual who is to be included in the database of authorized users must first be enrolled in the system. This is the analogous of assigning a password to a user.
  - ✓ the user presents a name and typically some type of passwords or PIN
  - ✓ the system senses some biometric characteristic of the user (ex: fingerprint of right index)
  - ✓ the system digitalizes the input and then extracts a set of features that can be stored as a number or set of numbers, called template, and represent the unique biometric characteristics
  - ✓ the user is now enrolled in the system, which maintains for the user a name ID, a password or PIN, and the biometric value

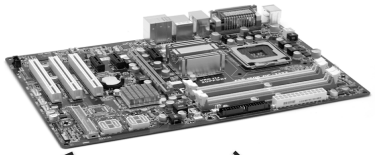
Biometric

Reader

Feature vector

Reference vector

Comparison algorithm

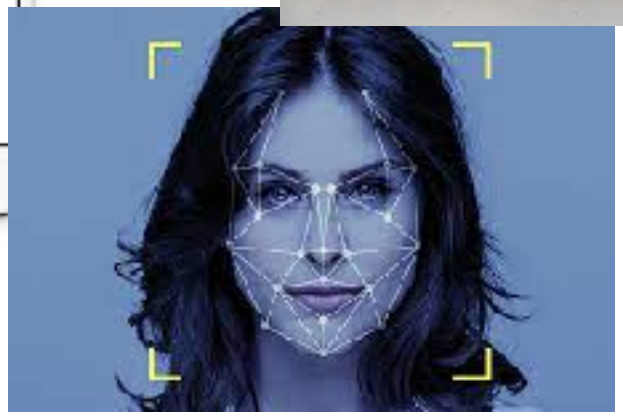


matches

doesn’t match

Physical Biometric

- A biometric authentication attempts to authenticate an individual based on his/her unique physical characteristics:
- Generally, biometric systems incorporate some sort of sensor or scanner to read in biometric information and then compare this information to stored templates of accepted users before granting access



Behavioural Biometric

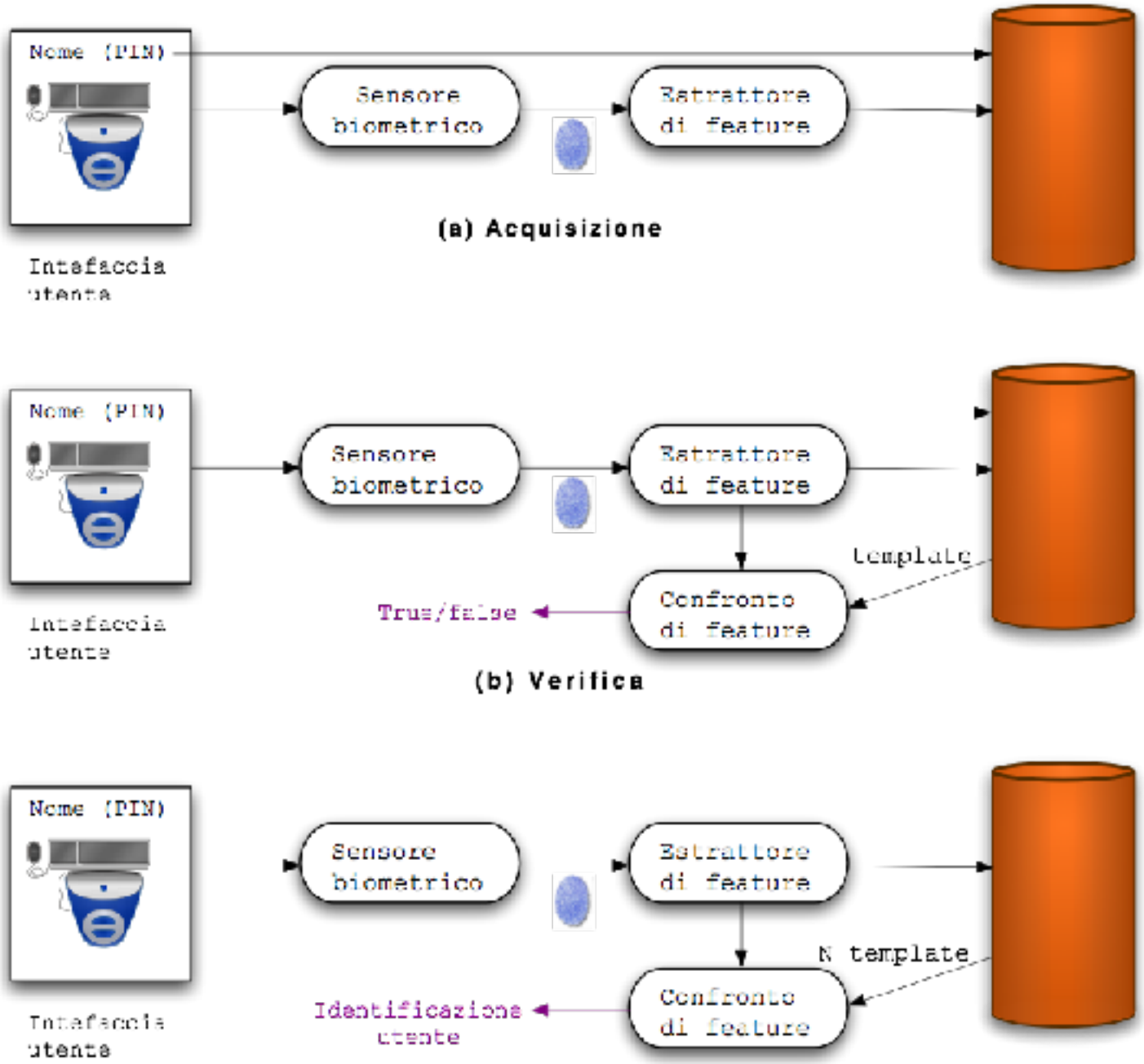
- **Typing Style:** The speed and the amount of time a person takes from one letter to another, the rhythm in which individual type characters on a keyboard or keypad, the degree of impact on the keyboard, etc. are major biometrics for authentication.
- **Navigation Style:** Finger movement, keyboard movement, scrolling movement, mouse movement on the screens & computer are generally considered as navigation style biometrics. If the movement is detected to be different than stored data, the system will lock the device immediately.
- **Interaction Style:** Technology is connecting people, but all in a different way. Apps are to be used by all of us, but which app and at what time and for how much time we use is not the same, right? How often we go to a particular app and how much time spent on social media is actually the means of identifying a person's behavior.
- **Physical Style:** How do you tilt your phone when you hold it? How do you walk with the phone? When the company has to provide high data protection authorities in critical areas, certain behavioral biometric elements can be a life savior.

Chemical and Vein Biometric

- **DNA:** Such scanners are used by law enforcement to identify criminals. However, this was slow at the beginning, but now it has come out to be rendered a DNA data match in minutes at a reasonable cost.
- **Vein Recognition:** It is the type that can be used to classify individuals in the human finger or palm based on the vein patterns.

Operations of a biometric authentication

- Verification: is analogous to a user logging on to a system by using a memory card or smart card coupled with a password or PIN.
  - ✓ For biometric verification, the user enters a PIN and also uses a biometric sensor.
  - ✓ The system extracts the corresponding feature and compares that to the template stored for this user.
  - ✓ If there is a match, then the system authenticates this user
- Identification: the individual uses the biometric sensor but presents no additional information. The system then compares the resend template with the set of stored templates, If there is a match the user is identified, otherwise the user is rejected.

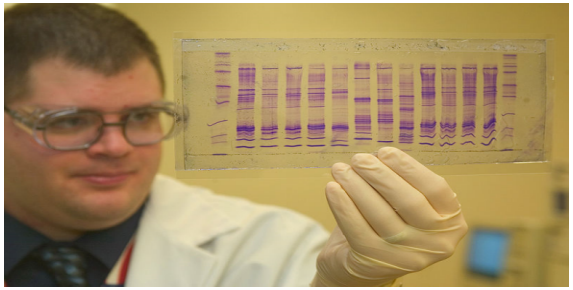

Nome (PIN)
Sensore biometrico
Estrattore di feature
Interfaccia utente
(a) Acquisizione
Nome (PIN)
Sensore biometrico
Estrattore di feature
Template
True/false
Confronto di feature
Interfaccia utente
(b) Verifica
Nome (PIN)
Sensore biometrico
Estrattore di feature
N template
Identificazione utente
Confronto di feature
Interfaccia utente
(c) Identificazione

Candidates for biometric authentication

- Signature
- Fingerprints
- Retinal/iris scans
- DNA
- “Blue-ink” signature
- Voice recognition
- Face recognition
- Gait recognition

Public domain image from
http://commons.wikimedia.org/wiki/File:Fingerprint_Arch.jpg

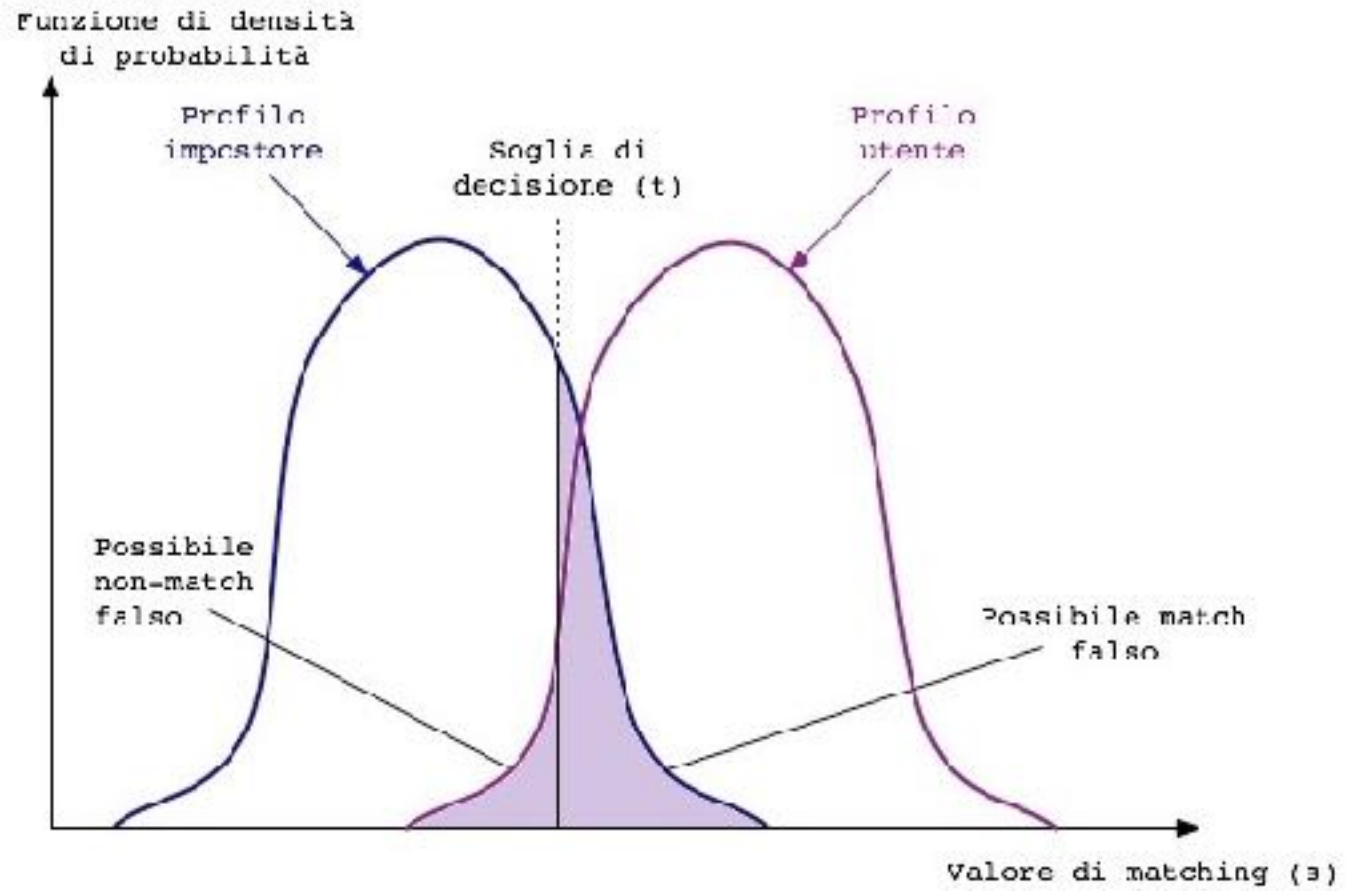
Public domain image from
http://commons.wikimedia.org/wiki/File:Retinal_scan_securimetrics.jpg



Public domain image from
http://commons.wikimedia.org/wiki/File:CBP_chemist_reads_a_DNA_profile.jpg

Biometric Accuracy

Given the complexity of the physical features it is not possible to have a perfect correspondence between what is stored and what is presented for authentication



Problems with biometrics

Limitations

- Accuracy of matching algorithm
  - ✓ False positives: allow access to unauthorized user
  - ✓ False negatives: reject a legitimate user
- Easy forging of biometric traits
  - ✓ Fingerprints left in many places
- Low user acceptance
  - ✓ User may not like to have their retina scanned

Advantages	Disadvantages
✔ Easy to Use A fingerprint or iris scan is much easier to use than a long and irritating password. Detecting a fingerprint and allowing a person to open the phone just takes a second for all modern smartphones. **✔ Not Transferable** You can't transfer your fingers to someone else like a password, until and unless you cut them! It requires a present authorization and a physical application being the only way to access most biometric authentication schemes. So rest assured that no one can hack your device, except it is you! **✔ No Spoofing** As it is not possible to transfer your fingers, it is equally impossible to reproduce biometrics such as facial markings, fingerprints, iris scanning even with advanced technologies. Also, do you think your fingerprints can match with someone? In fact, you have more shot to win a lottery than having the same fingerprint as a hacker seeking to get into your account that is secured by biometrics. **✔ Great User Experience** Although the internal mechanisms for biometric authentication are technical, it is extremely simpler and quicker from a user's point of view than password protections. Also, losing a password is a common error for the majority of users. Are there any chances that you put your biometrics somewhere and forgot it? Maybe in dreams!	**✘ High Cost** We know that the more complex defense framework would ask for more considerable investment and expense and it is no shock. But, most users also need to update existing technologies to accommodate a change to biometric authentication on their smartphones. So that can be a concern. **✘ Breach of Data** Businesses and governments that gather and retain personal biometric data from users are under persistent hacker attacks. Since biometric data is irreplaceable, companies need to handle confidential biometric data with enhanced vigilance and care in order to stay ahead of fraud developments. **✘ False Positives** Biometric authentication factors rely on partial knowledge to authenticate the identity of a person. For example, during the enrollment process, mobile biometric systems can scan an entire fingerprint and translate it into data. Future biometric fingerprint verification, though, can only use portions of the prints to check identities, so it can give false positives sometimes.

How biometrics are attacked

- Presentation attacks involve an impostor using an artefact of some kind to mimic an individual who *has* been enrolled in the system

- Sensor output interception: an attacker may seek to modify or intercept the data output from the sensor. A previously captured sample might be replayed, or a captured biometric sample could be substituted with biometric data of a different individual at enrolment.

- Reference and database-related vulnerabilities: an attacker may target data during transmission, or in storage by the biometric system.

- Integrity of enrolment: There is a possibility that the enrolment process could be subverted, allowing the acceptance of inappropriate enrolment data.

- Denial of service attacks: ll systems are vulnerable to denial of service attacks. In the case of a biometric system, this will divert subjects to the exception handling system. It is therefore important that this fallback system is no less secure than the biometric system.

- Insider threat: All security systems are vulnerable to an attack by a trusted system administrator or operator. Due to the level of access and trust held by such people, insider attacks on a biometric system can take any of the forms outlined above.