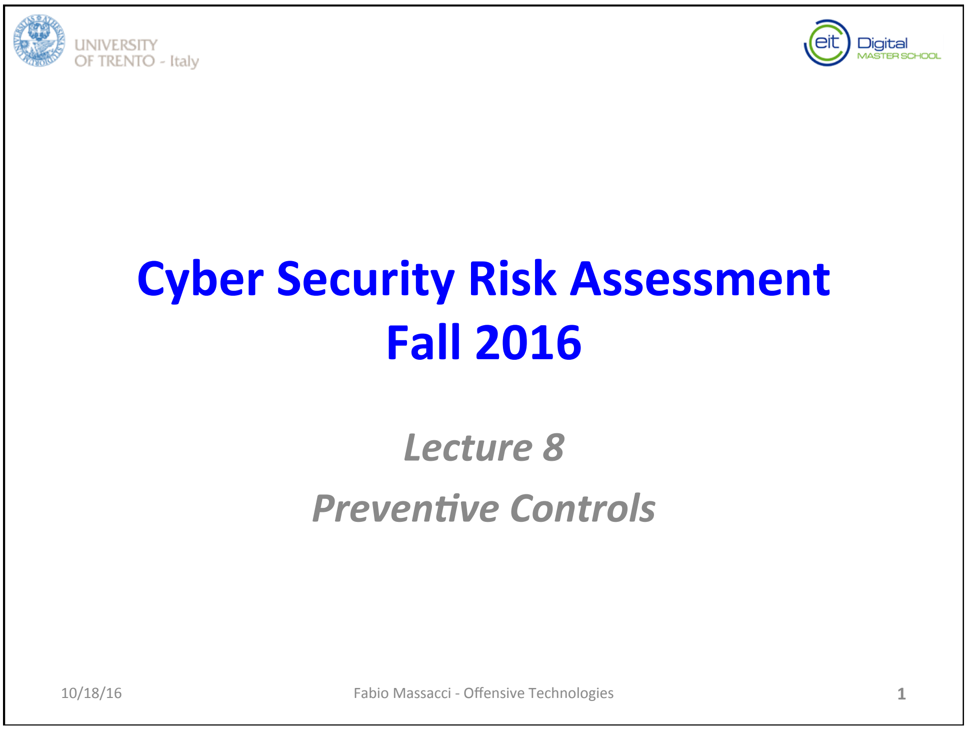
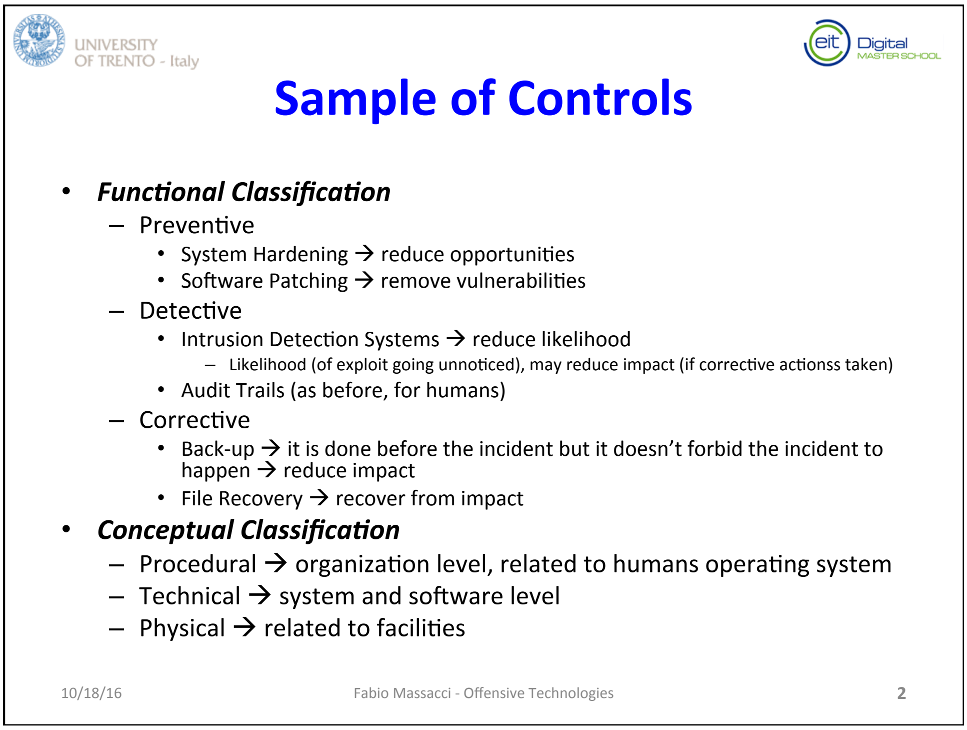


Cyber Security Risk Assessment Fall 2016

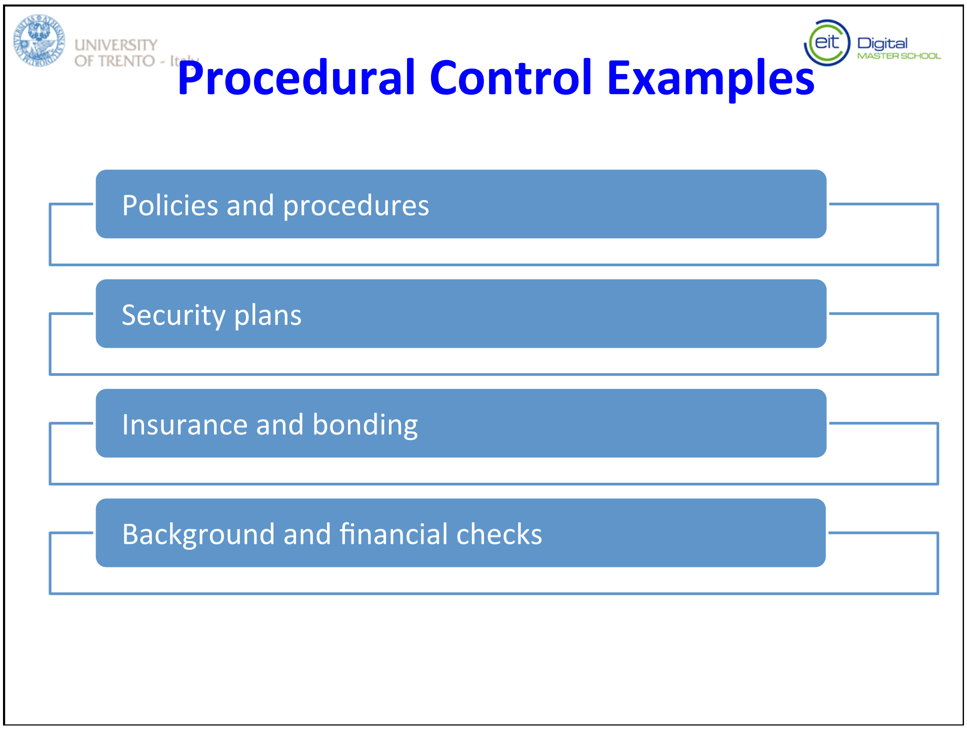
Lecture 8

Preventive Controls



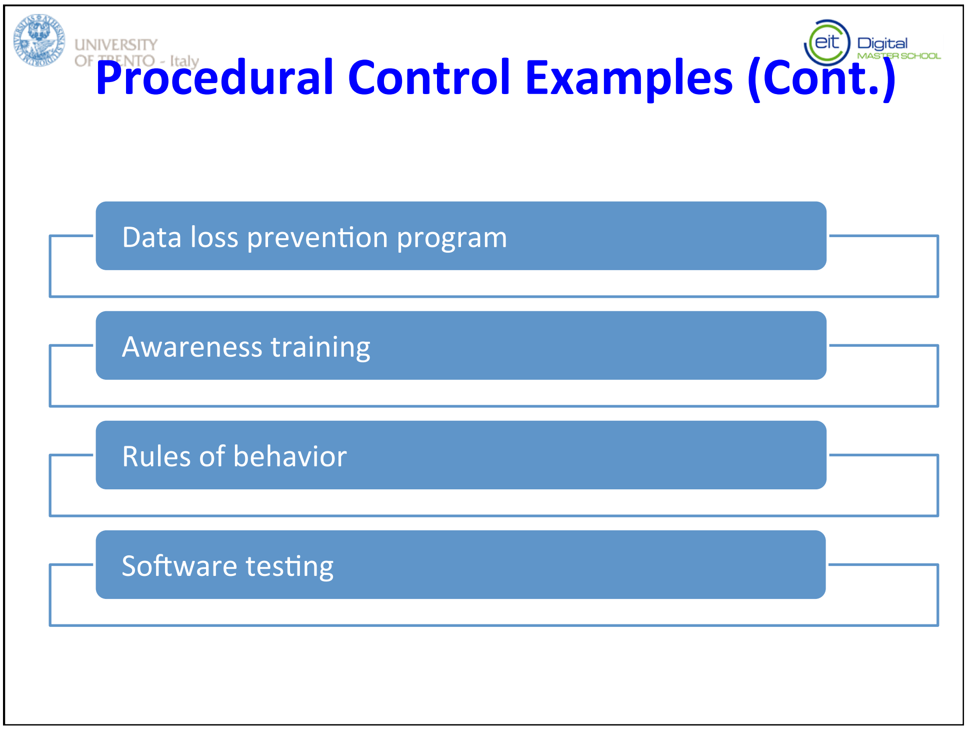
Sample of Controls

- ***Functional Classification***
  - Preventive
    - System Hardening → reduce opportunities
    - Software Patching → remove vulnerabilities
  - Detective
    - Intrusion Detection Systems → reduce likelihood
      - Likelihood (of exploit going unnoticed), may reduce impact (if corrective actionss taken)
    - Audit Trails (as before, for humans)
  - Corrective
    - Back-up → it is done before the incident but it doesn't forbid the incident to happen → reduce impact
    - File Recovery → recover from impact
- ***Conceptual Classification***
  - Procedural → organization level, related to humans operating system
  - Technical → system and software level
  - Physical → related to facilities



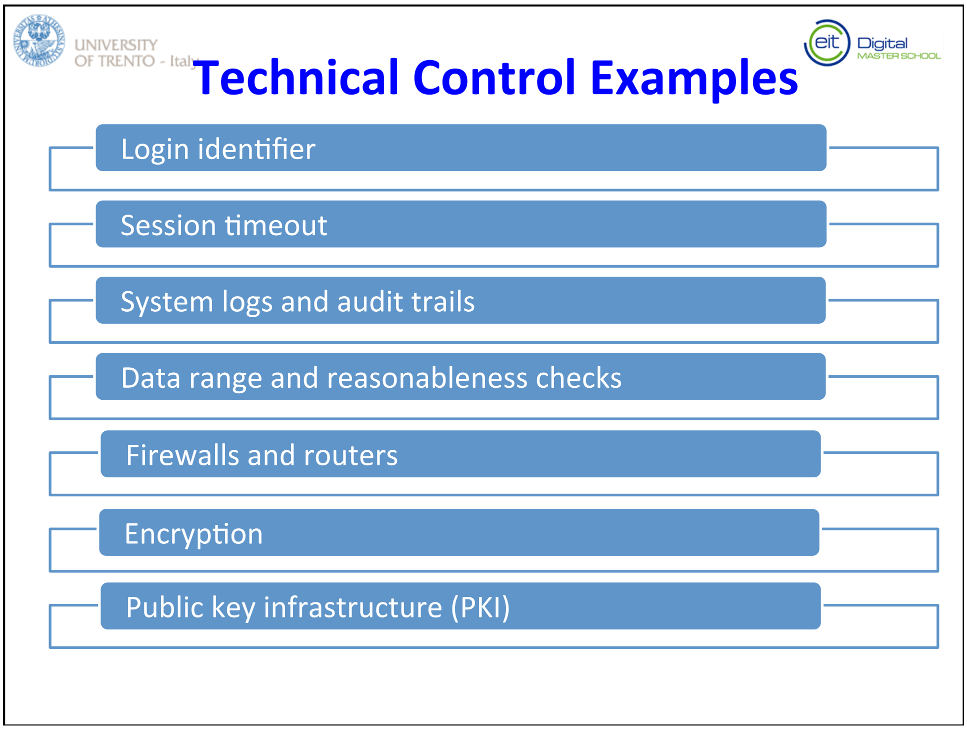
Procedural Control Examples

- Policies and procedures
- Security plans
- Insurance and bonding
- Background and financial checks



Procedural Control Examples (Cont.)

- Data loss prevention program
- Awareness training
- Rules of behavior
- Software testing

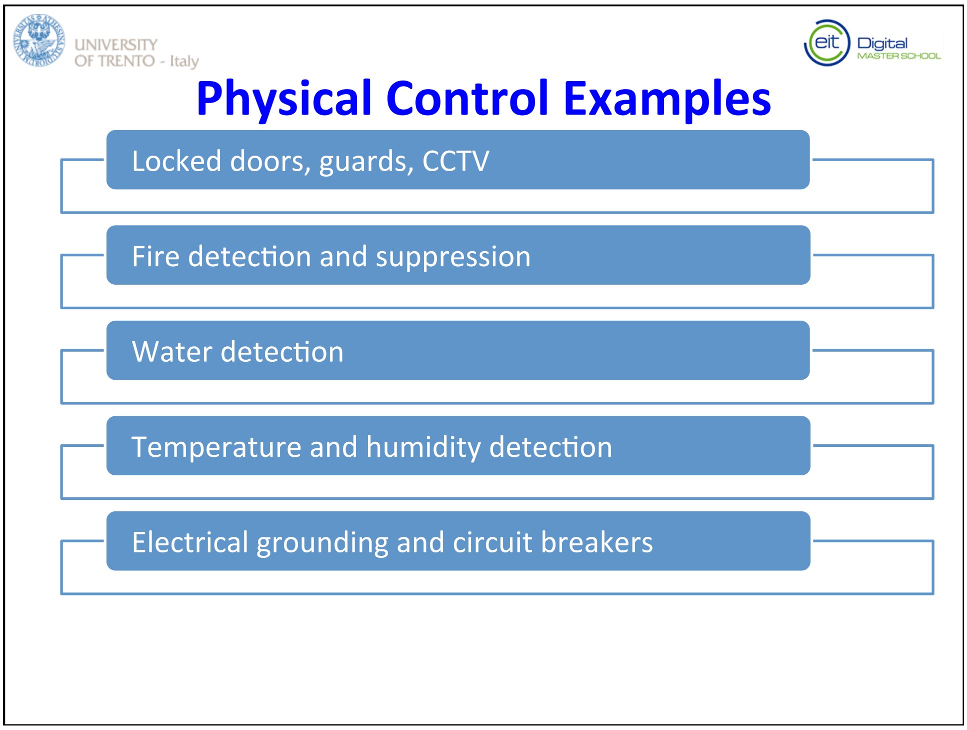


UNIVERSITY OF TRENTO - Ital

eit Digital MASTER SCHOOL

Technical Control Examples

- Login identifier
- Session timeout
- System logs and audit trails
- Data range and reasonableness checks
- Firewalls and routers
- Encryption
- Public key infrastructure (PKI)

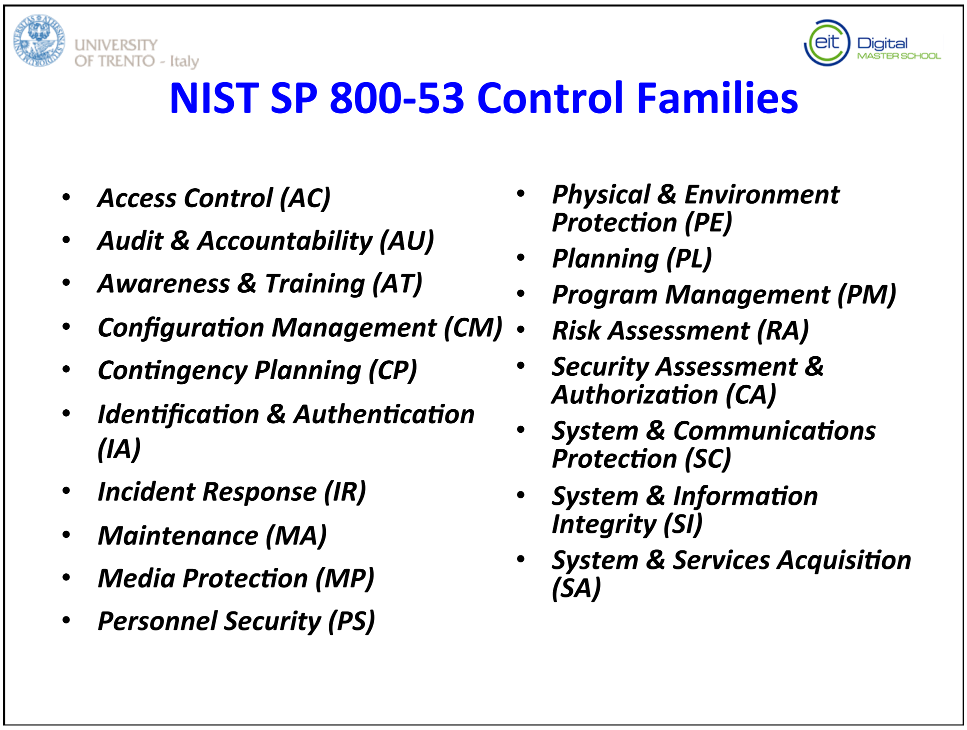


UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Physical Control Examples

- Locked doors, guards, CCTV
- Fire detection and suppression
- Water detection
- Temperature and humidity detection
- Electrical grounding and circuit breakers

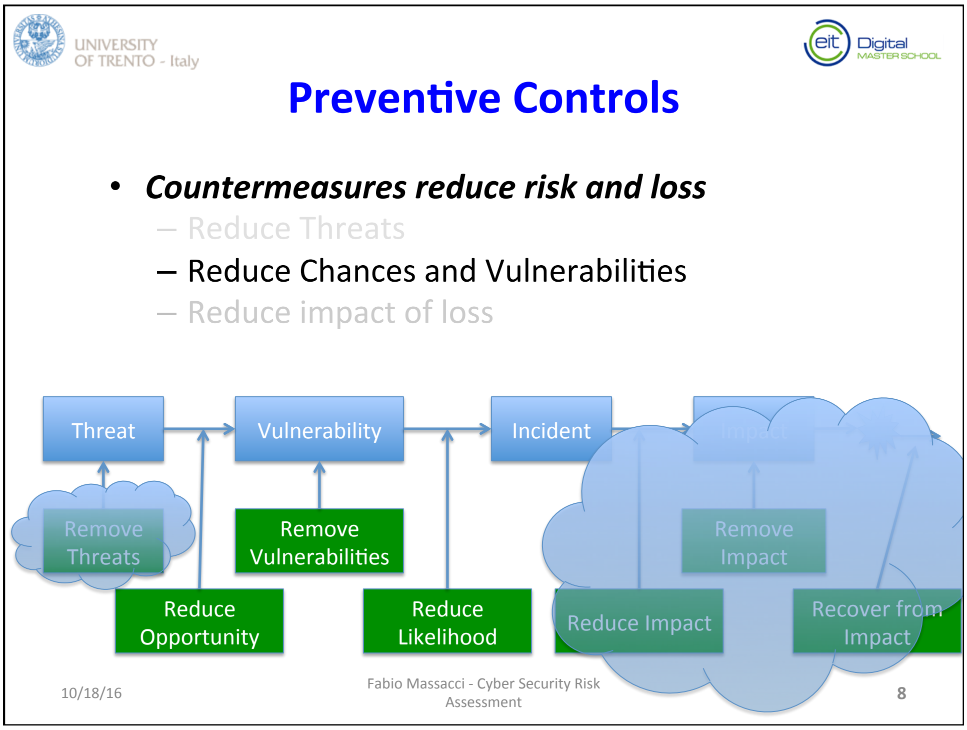


UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

NIST SP 800-53 Control Families

- *Access Control (AC)*
- *Audit & Accountability (AU)*
- *Awareness & Training (AT)*
- *Configuration Management (CM)*
- *Contingency Planning (CP)*
- *Identification & Authentication (IA)*
- *Incident Response (IR)*
- *Maintenance (MA)*
- *Media Protection (MP)*
- *Personnel Security (PS)*
- *Physical & Environment Protection (PE)*
- *Planning (PL)*
- *Program Management (PM)*
- *Risk Assessment (RA)*
- *Security Assessment & Authorization (CA)*
- *System & Communications Protection (SC)*
- *System & Information Integrity (SI)*
- *System & Services Acquisition (SA)*



UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Preventive Controls

- ***Countermeasures reduce risk and loss***
  - Reduce Threats
  - Reduce Chances and Vulnerabilities
  - Reduce impact of loss

Threat

Vulnerability

Incident

Impact

Remove Threats

Remove Vulnerabilities

Remove Impact

Reduce Opportunity

Reduce Likelihood

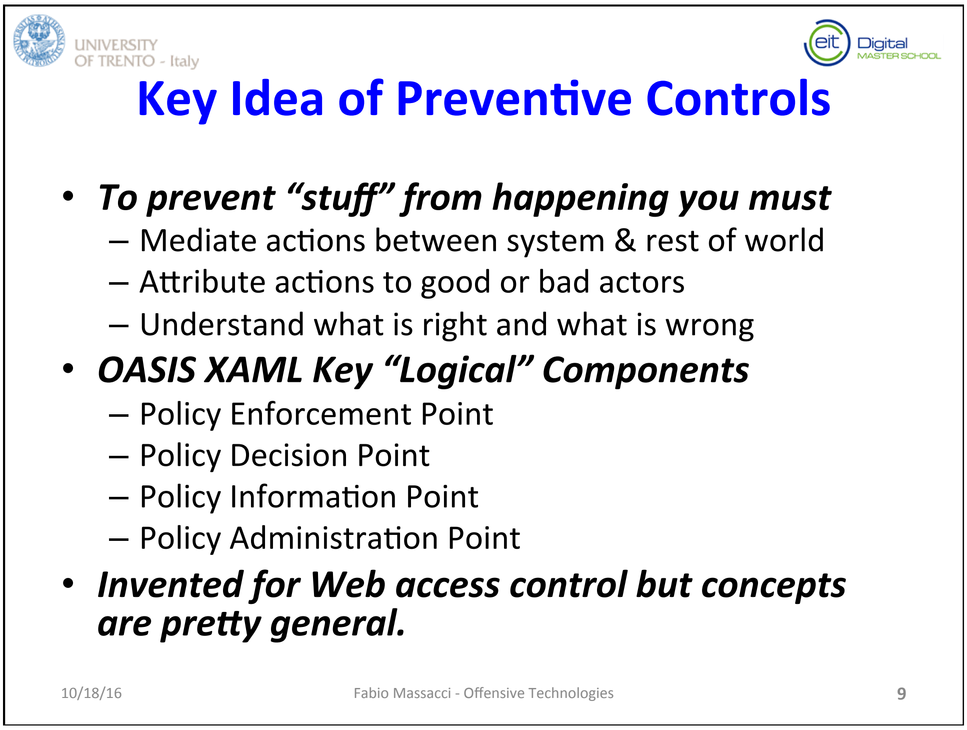
Reduce Impact

Recover from Impact

10/18/16

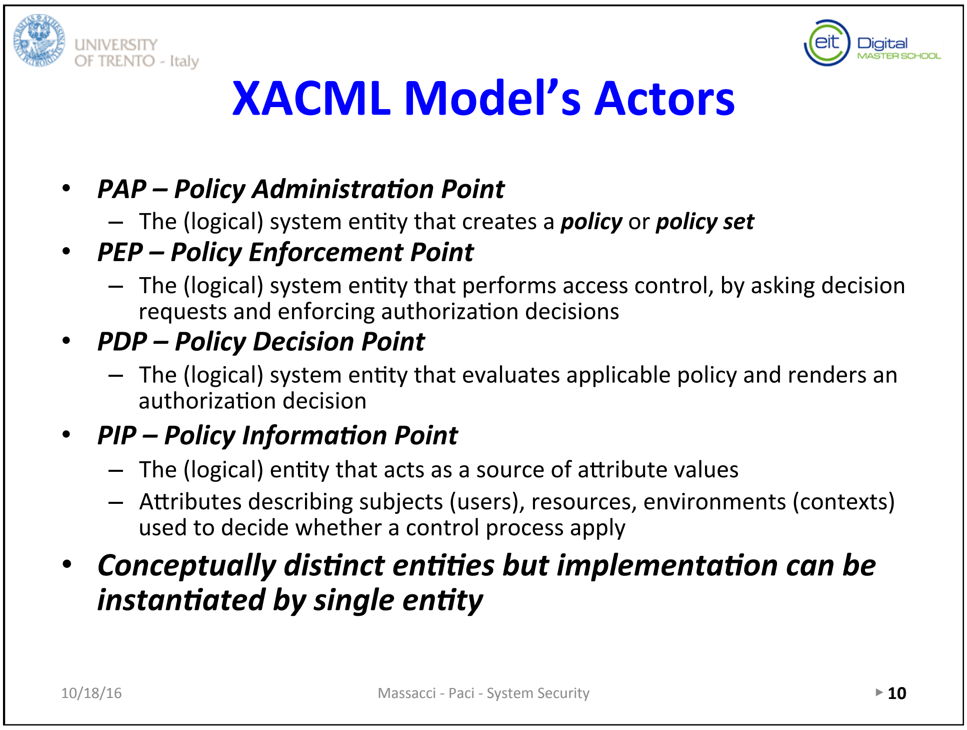
Fabio Massacci - Cyber Security Risk Assessment

8



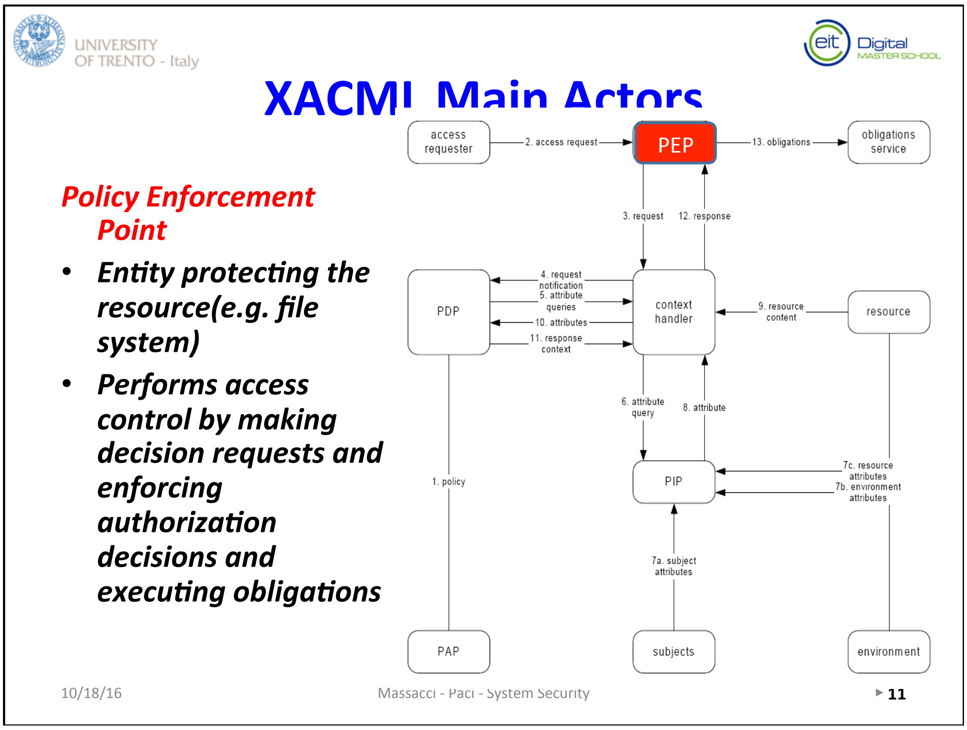
Key Idea of Preventive Controls

- ***To prevent "stuff" from happening you must***
  - Mediate actions between system & rest of world
  - Attribute actions to good or bad actors
  - Understand what is right and what is wrong
- ***OASIS XAML Key "Logical" Components***
  - Policy Enforcement Point
  - Policy Decision Point
  - Policy Information Point
  - Policy Administration Point
- ***Invented for Web access control but concepts are pretty general.***



XACML Model's Actors

- ***PAP – Policy Administration Point***
  - The (logical) system entity that creates a ***policy*** or ***policy set***
- ***PEP – Policy Enforcement Point***
  - The (logical) system entity that performs access control, by asking decision requests and enforcing authorization decisions
- ***PDP – Policy Decision Point***
  - The (logical) system entity that evaluates applicable policy and renders an authorization decision
- ***PIP – Policy Information Point***
  - The (logical) entity that acts as a source of attribute values
  - Attributes describing subjects (users), resources, environments (contexts) used to decide whether a control process apply
- ***Conceptually distinct entities but implementation can be instantiated by single entity***



UNIVERSITY OF TRENTO - Italy

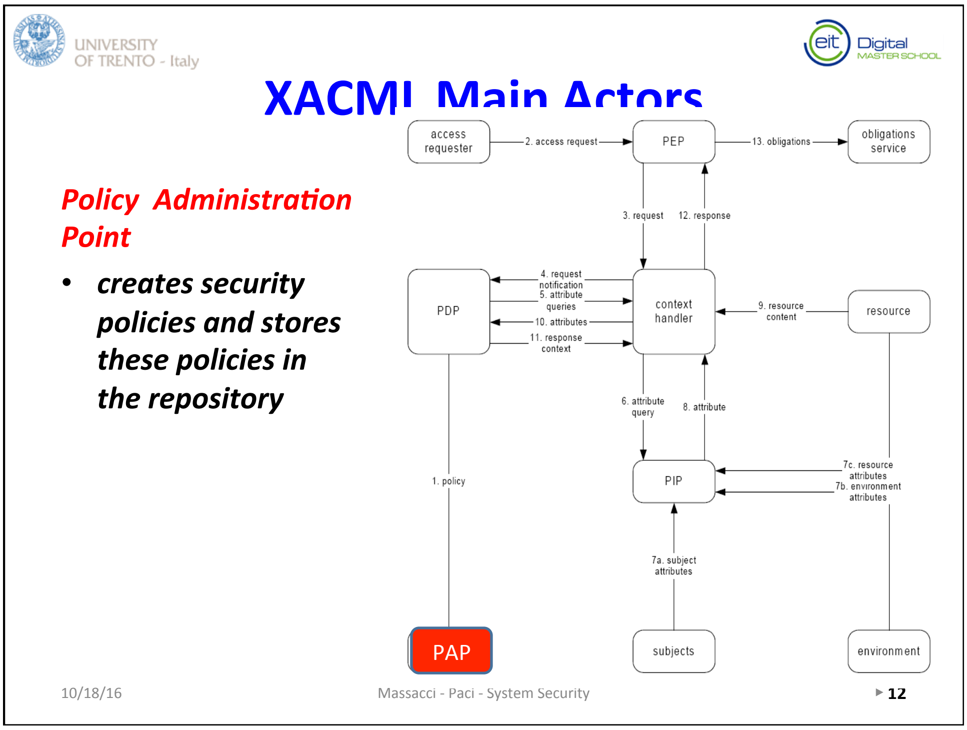
eit Digital MASTER SCHOOL

XACML Main Actors

Policy Enforcement Point

- ***Entity protecting the resource(e.g. file system)***
- ***Performs access control by making decision requests and enforcing authorization decisions and executing obligations***

10/18/16 Massacci - Paci - System Security 11



UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

XACML Main Actors

Policy Administration Point

- ***creates security policies and stores these policies in the repository***

10/18/16 Massacci - Paci - System Security 12

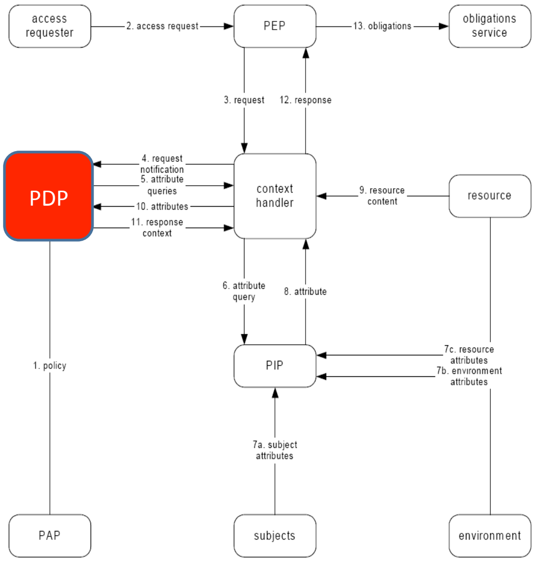
UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

XACML Main Actors

The Policy Decision Point

- ***Receives and examines the request***
- ***Retrieves applicable policies***
- ***evaluates the applicable policy and***
- ***Returns the authorization decision to PEP***



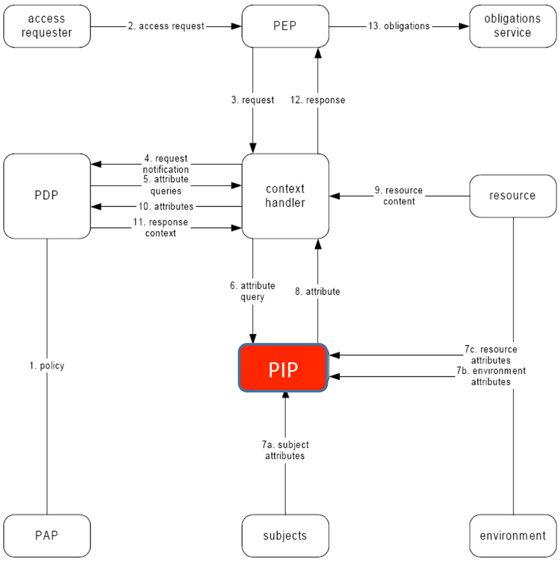
UNIVERSITY OF TRENTO - Italy

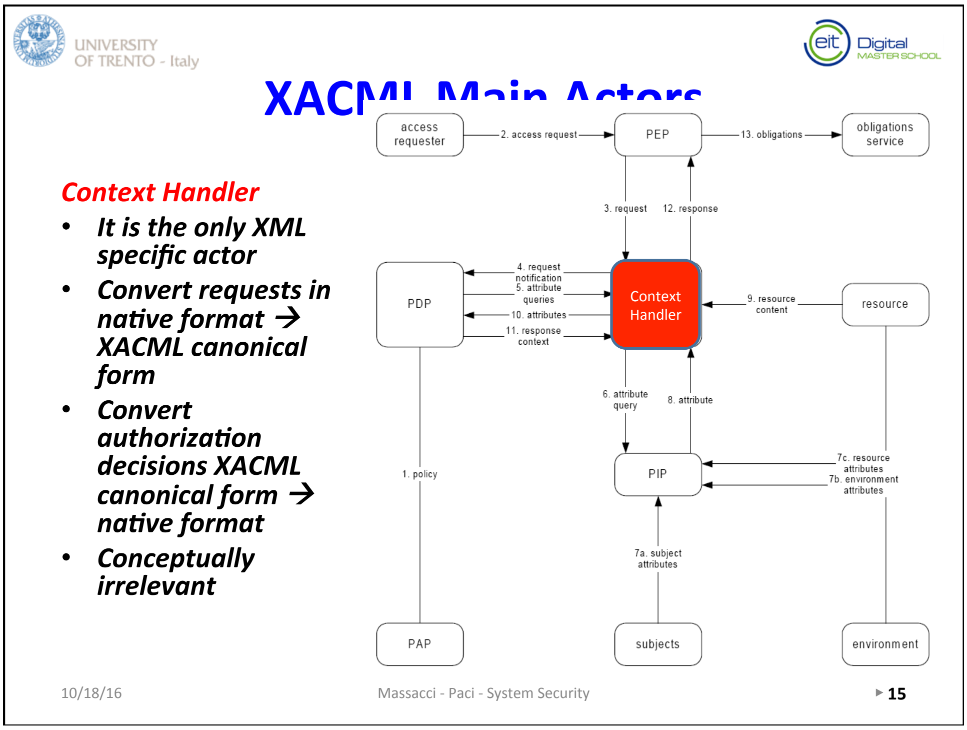
eit Digital MASTER SCHOOL

XACML Main Actors

Policy Information Point

- ***serves as the source of attribute values, or the data required for policy evaluation***





UNIVERSITY OF TRENTO - Italy

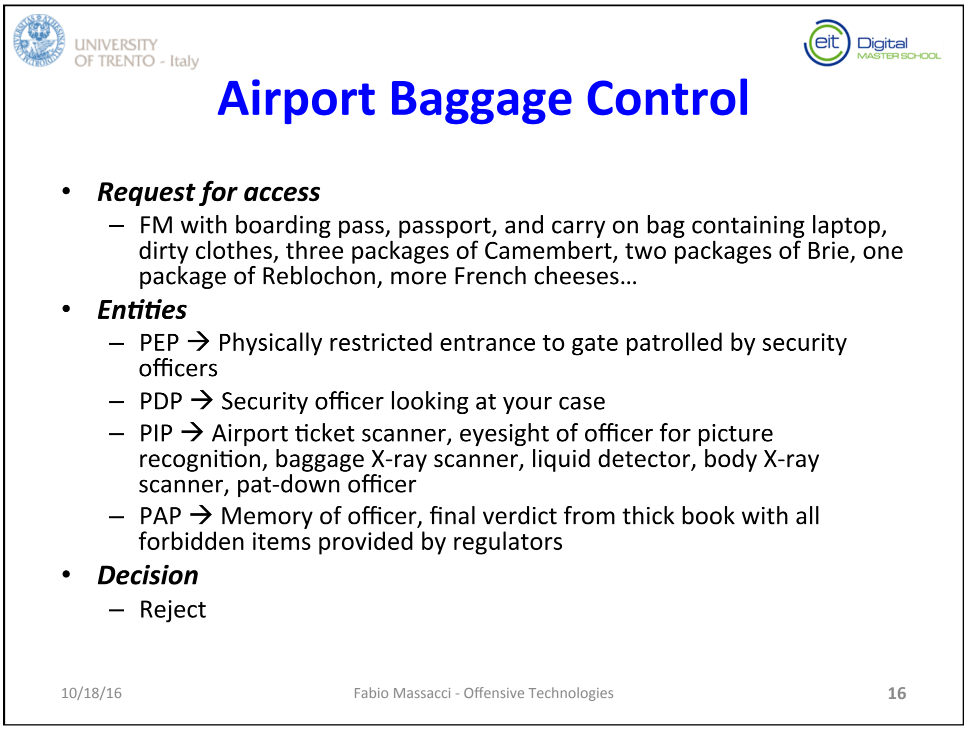
eit Digital MASTER SCHOOL

XACML Main Actors

Context Handler

- ***It is the only XML specific actor***
- ***Convert requests in native format → XACML canonical form***
- ***Convert authorization decisions XACML canonical form → native format***
- ***Conceptually irrelevant***

10/18/16 Massacci - Paci - System Security 15



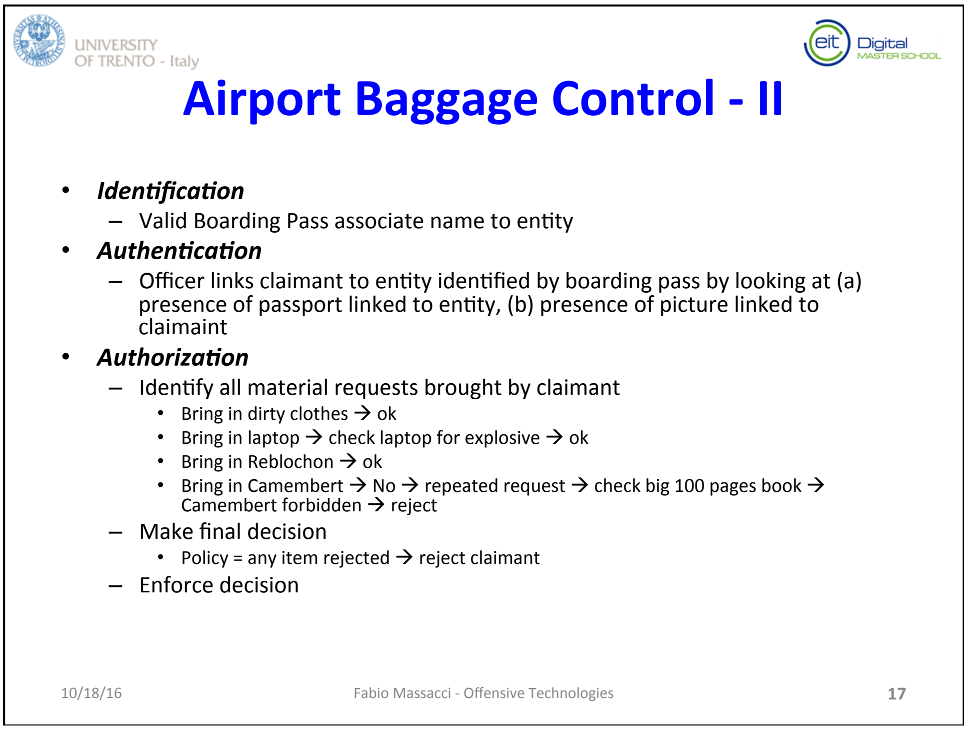
UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Airport Baggage Control

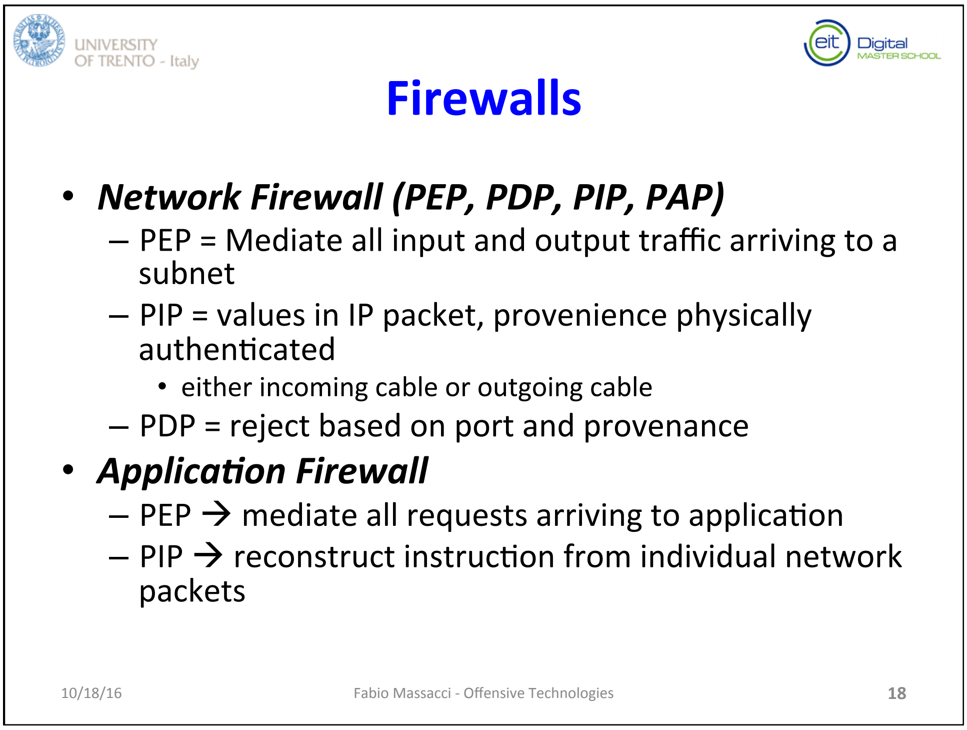
- ***Request for access***
  - FM with boarding pass, passport, and carry on bag containing laptop, dirty clothes, three packages of Camembert, two packages of Brie, one package of Reblochon, more French cheeses…
- ***Entities***
  - PEP → Physically restricted entrance to gate patrolled by security officers
  - PDP → Security officer looking at your case
  - PIP → Airport ticket scanner, eyesight of officer for picture recognition, baggage X-ray scanner, liquid detector, body X-ray scanner, pat-down officer
  - PAP → Memory of officer, final verdict from thick book with all forbidden items provided by regulators
- ***Decision***
  - Reject

10/18/16 Fabio Massacci - Offensive Technologies 16



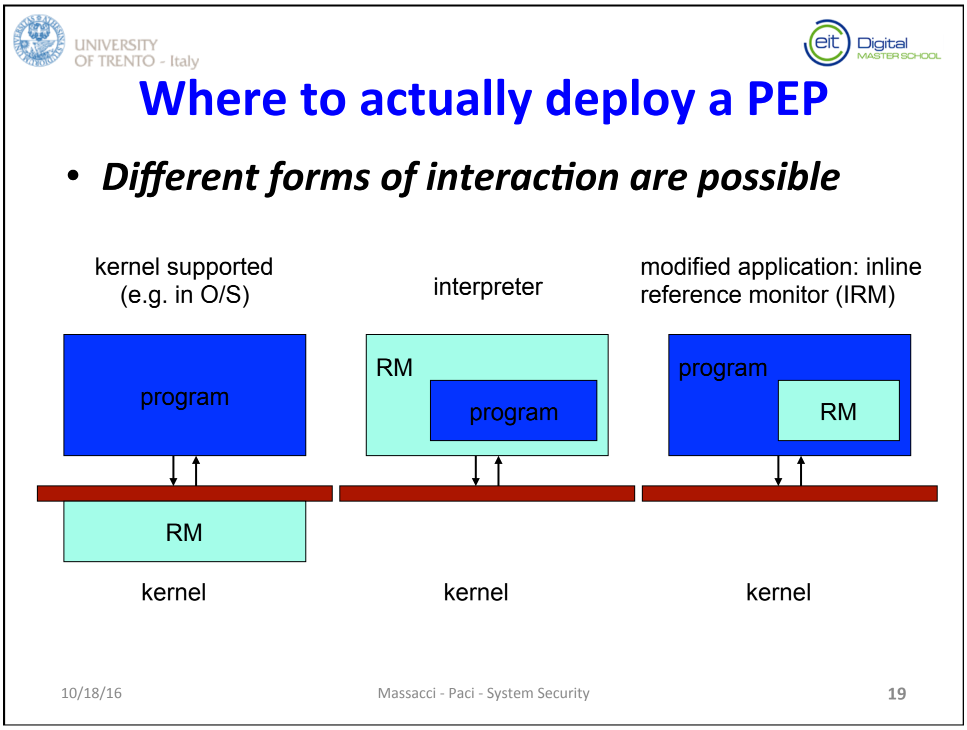
Airport Baggage Control - II

- ***Identification***
  - Valid Boarding Pass associate name to entity
- ***Authentication***
  - Officer links claimant to entity identified by boarding pass by looking at (a) presence of passport linked to entity, (b) presence of picture linked to claimaint
- ***Authorization***
  - Identify all material requests brought by claimant
    - Bring in dirty clothes → ok
    - Bring in laptop → check laptop for explosive → ok
    - Bring in Reblochon → ok
    - Bring in Camembert → No → repeated request → check big 100 pages book → Camembert forbidden → reject
  - Make final decision
    - Policy = any item rejected → reject claimant
  - Enforce decision



Firewalls

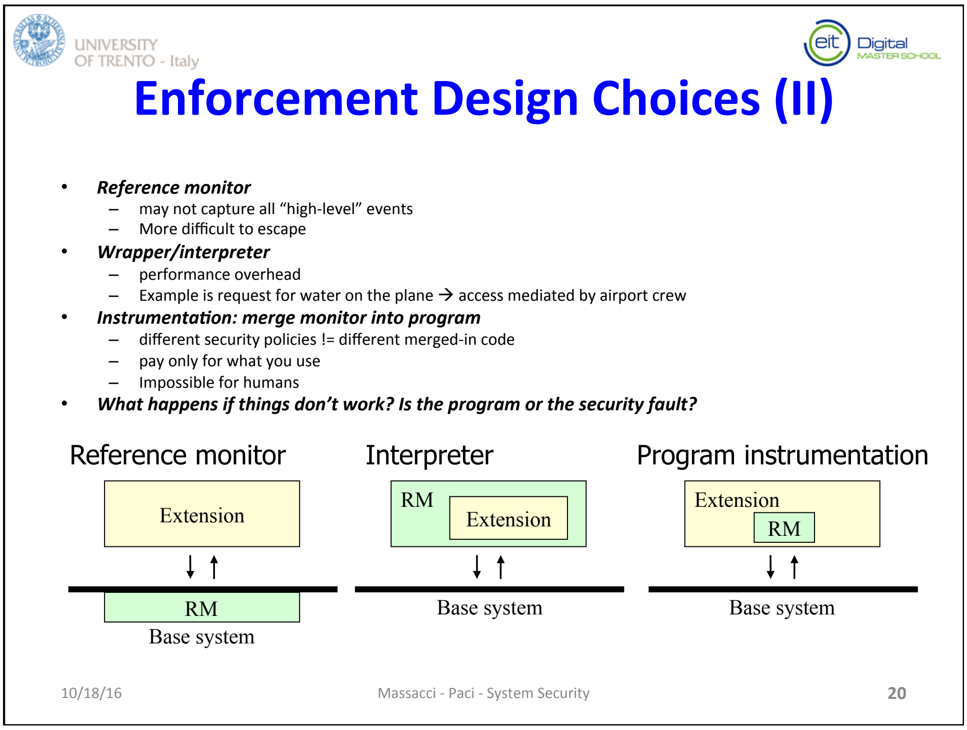
- ***Network Firewall (PEP, PDP, PIP, PAP)***
  - PEP = Mediate all input and output traffic arriving to a subnet
  - PIP = values in IP packet, provenience physically authenticated
    - either incoming cable or outgoing cable
  - PDP = reject based on port and provenance
- ***Application Firewall***
  - PEP → mediate all requests arriving to application
  - PIP → reconstruct instruction from individual network packets



Where to actually deploy a PEP

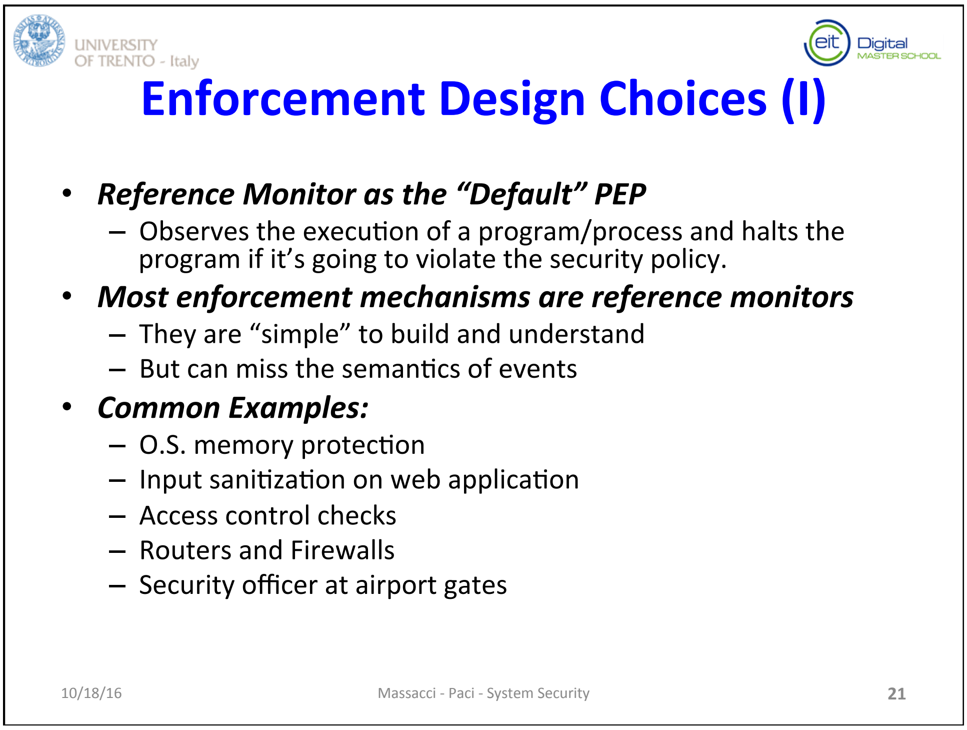
- ***Different forms of interaction are possible***

kernel supported (e.g. in O/S)
program
RM
kernel
interpreter
RM
program
kernel
modified application: inline reference monitor (IRM)
program
RM
kernel



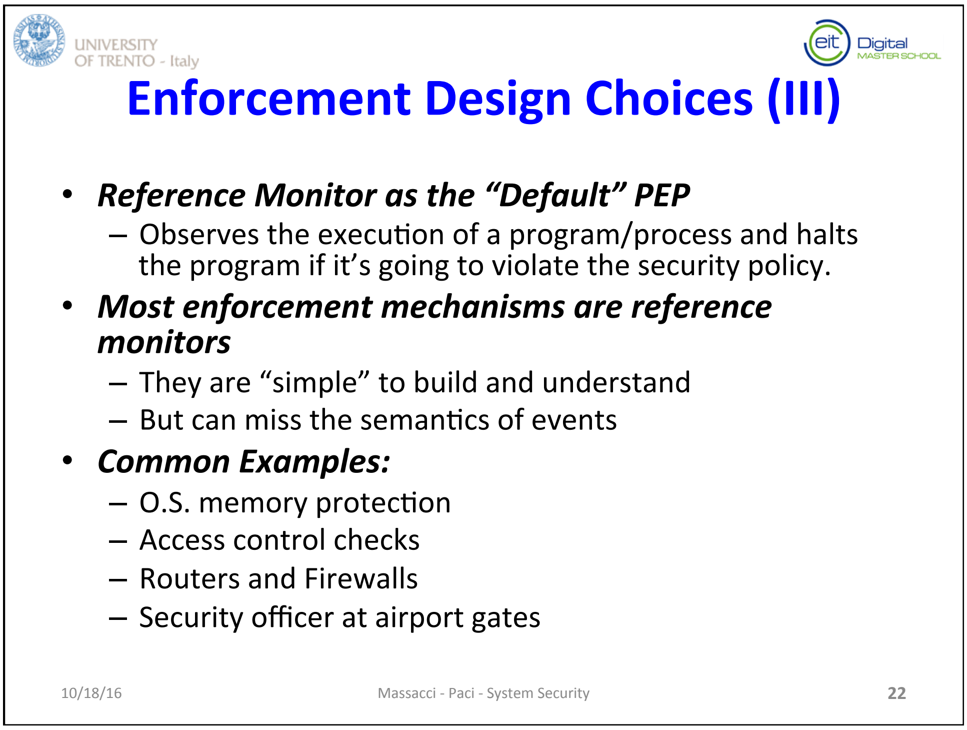
Enforcement Design Choices (II)

- ***Reference monitor***
  - may not capture all “high-level” events
  - More difficult to escape
- ***Wrapper/interpreter***
  - performance overhead
  - Example is request for water on the plane → access mediated by airport crew
- ***Instrumentation: merge monitor into program***
  - different security policies != different merged-in code
  - pay only for what you use
  - Impossible for humans
- ***What happens if things don’t work? Is the program or the security fault?***



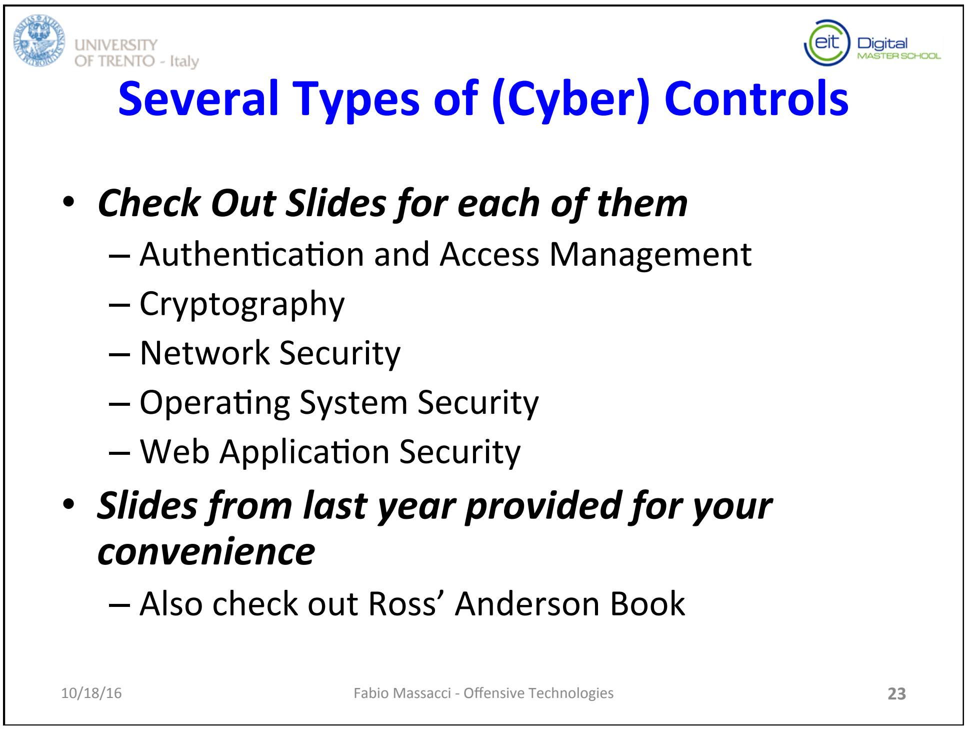
Enforcement Design Choices (I)

- ***Reference Monitor as the “Default” PEP***
  - Observes the execution of a program/process and halts the program if it’s going to violate the security policy.
- ***Most enforcement mechanisms are reference monitors***
  - They are “simple” to build and understand
  - But can miss the semantics of events
- ***Common Examples:***
  - O.S. memory protection
  - Input sanitization on web application
  - Access control checks
  - Routers and Firewalls
  - Security officer at airport gates



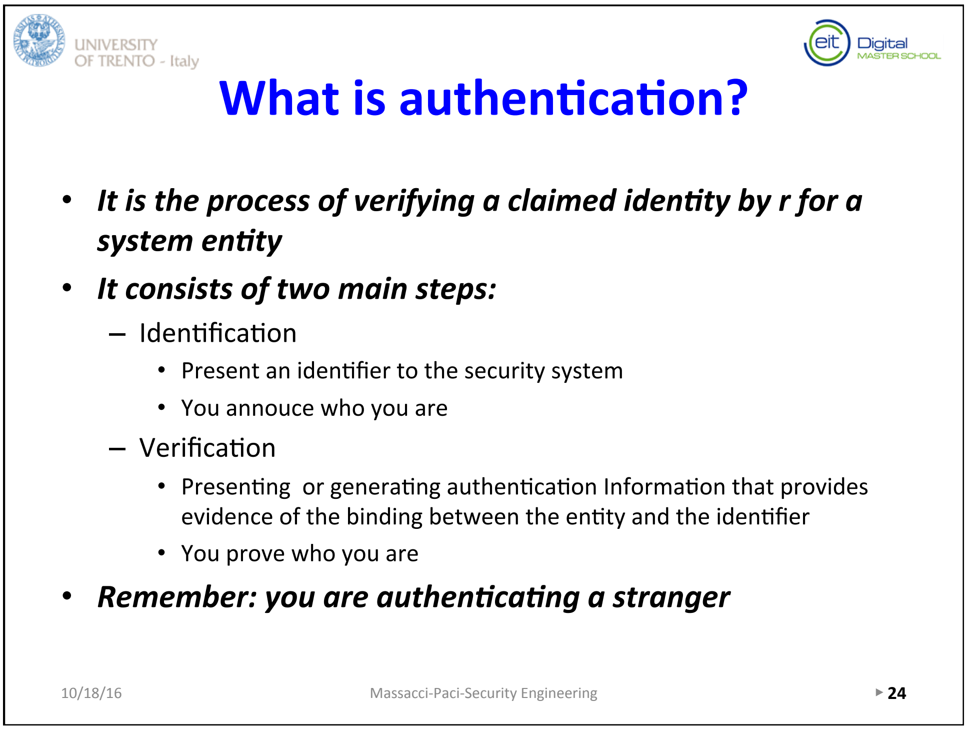
Enforcement Design Choices (III)

- ***Reference Monitor as the “Default” PEP***
  - Observes the execution of a program/process and halts the program if it’s going to violate the security policy.
- ***Most enforcement mechanisms are reference monitors***
  - They are “simple” to build and understand
  - But can miss the semantics of events
- ***Common Examples:***
  - O.S. memory protection
  - Access control checks
  - Routers and Firewalls
  - Security officer at airport gates



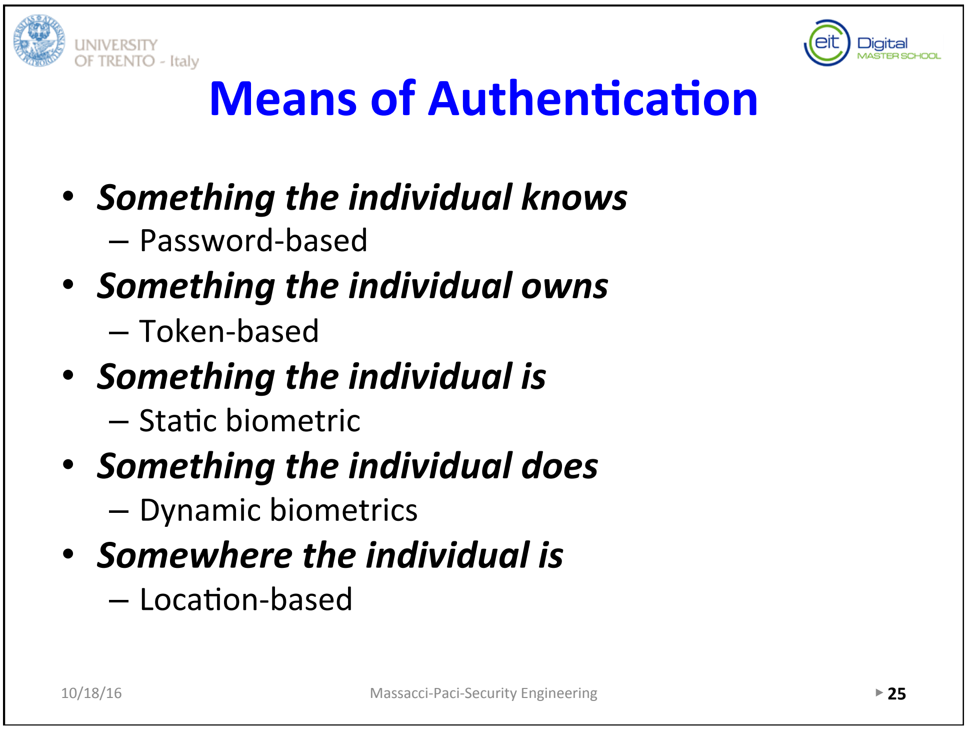
Several Types of (Cyber) Controls

- ***Check Out Slides for each of them***
  - Authentication and Access Management
  - Cryptography
  - Network Security
  - Operating System Security
  - Web Application Security
- ***Slides from last year provided for your convenience***
  - Also check out Ross' Anderson Book



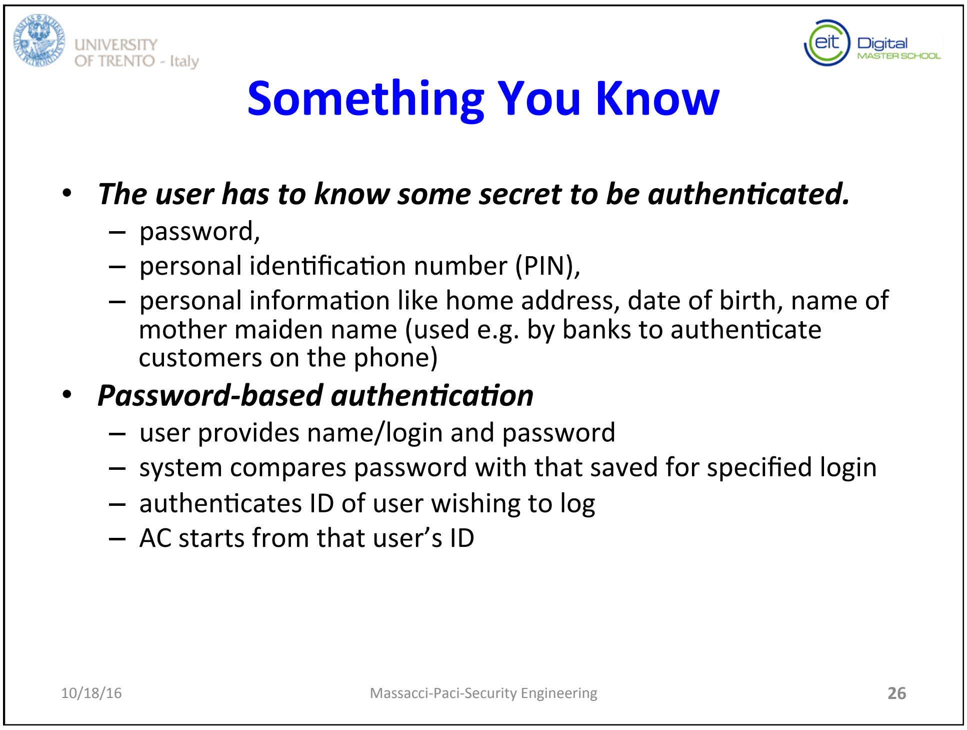
What is authentication?

- ***It is the process of verifying a claimed identity by r for a system entity***
- ***It consists of two main steps:***
  - Identification
    - Present an identifier to the security system
    - You annouce who you are
  - Verification
    - Presenting or generating authentication Information that provides evidence of the binding between the entity and the identifier
    - You prove who you are
- ***Remember: you are authenticating a stranger***



Means of Authentication

- ***Something the individual knows***
  – Password-based
- ***Something the individual owns***
  – Token-based
- ***Something the individual is***
  – Static biometric
- ***Something the individual does***
  – Dynamic biometrics
- ***Somewhere the individual is***
  – Location-based



Something You Know

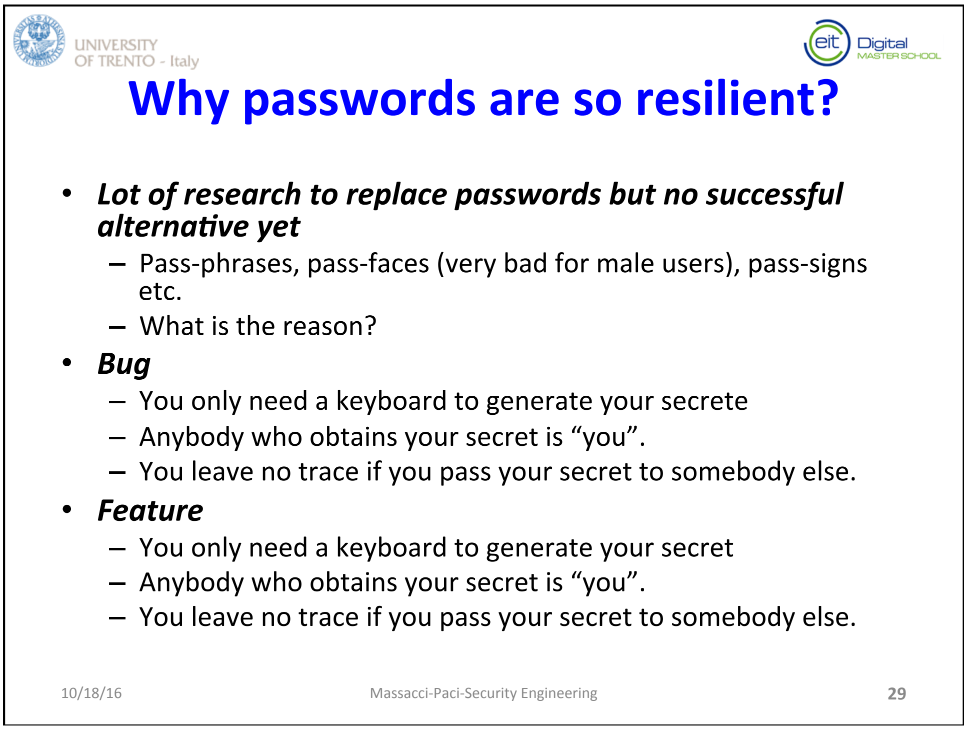
- ***The user has to know some secret to be authenticated.***
  – password,
  – personal identification number (PIN),
  – personal information like home address, date of birth, name of mother maiden name (used e.g. by banks to authenticate customers on the phone)
- ***Password-based authentication***
  – user provides name/login and password
  – system compares password with that saved for specified login
  – authenticates ID of user wishing to log
  – AC starts from that user's ID

Password Authentication

- ***Typical issues that need to be addressed:***
  - how to get the password to the user,
  - forgotten passwords,
  - password guessing,
  - protection of the password file
- ***Dangers***
  - User accounts without passwords.
  - Unchanged default passwords.
  - Badly chosen passwords – dictionary/brute force attacks.
  - Passwords stored in the clear.
  - Passwords transmitted in the clear.
  - Users forget passwords
    - the infrastructure for re-issuing passwords can be quite expensive (if it has to be truly secure)

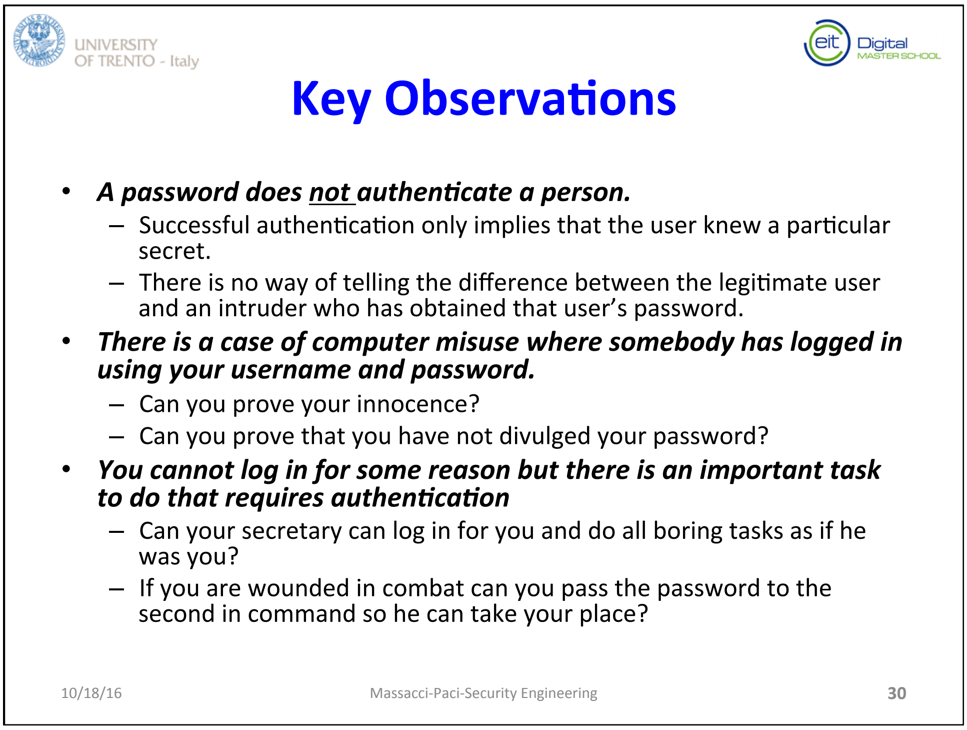
Why passwords are so resilient?

- ***Lot of research to replace passwords but no successful alternative yet***
  - Pass-phrases, pass-faces (very bad for male users), pass-signs etc.
  - What is the reason?
- ***Bugs***
  - .
  - .
- ***Features***
  - .
  - .
  - .



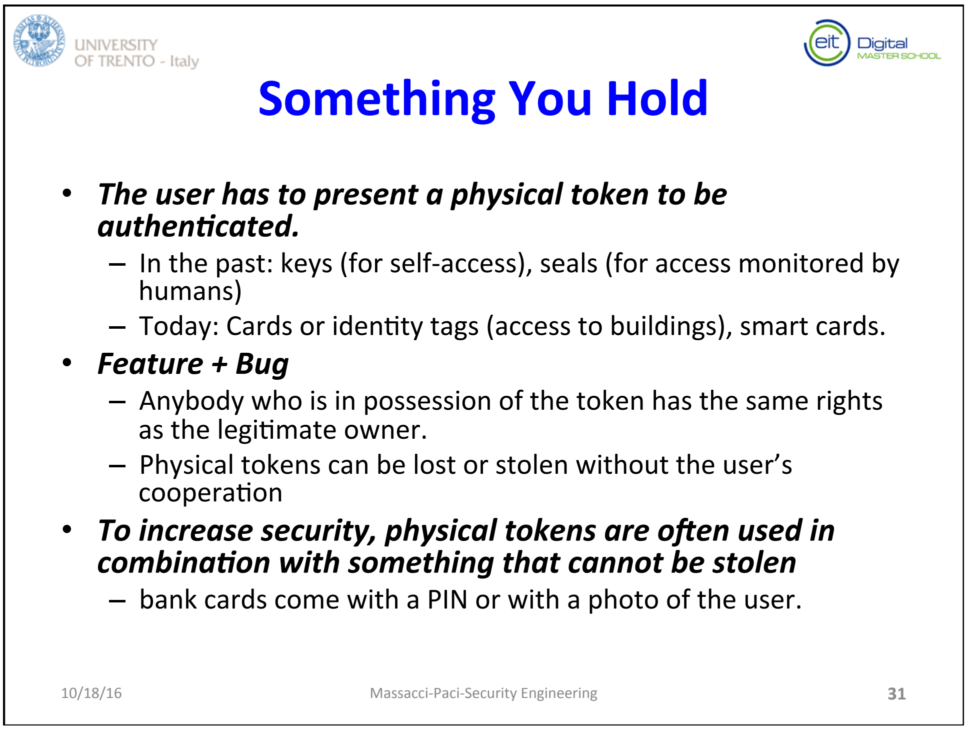
Why passwords are so resilient?

- ***Lot of research to replace passwords but no successful alternative yet***
  - Pass-phrases, pass-faces (very bad for male users), pass-signs etc.
  - What is the reason?
- ***Bug***
  - You only need a keyboard to generate your secrete
  - Anybody who obtains your secret is "you".
  - You leave no trace if you pass your secret to somebody else.
- ***Feature***
  - You only need a keyboard to generate your secret
  - Anybody who obtains your secret is "you".
  - You leave no trace if you pass your secret to somebody else.



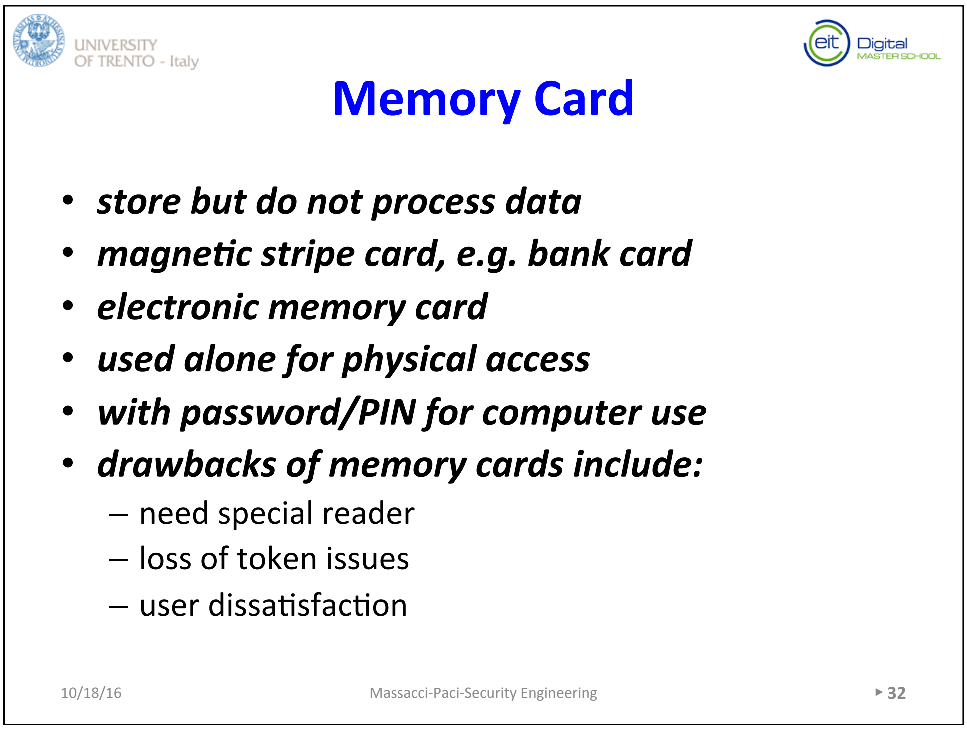
Key Observations

- ***A password does not authenticate a person.***
  - Successful authentication only implies that the user knew a particular secret.
  - There is no way of telling the difference between the legitimate user and an intruder who has obtained that user's password.
- ***There is a case of computer misuse where somebody has logged in using your username and password.***
  - Can you prove your innocence?
  - Can you prove that you have not divulged your password?
- ***You cannot log in for some reason but there is an important task to do that requires authentication***
  - Can your secretary can log in for you and do all boring tasks as if he was you?
  - If you are wounded in combat can you pass the password to the second in command so he can take your place?



Something You Hold

- ***The user has to present a physical token to be authenticated.***
  - In the past: keys (for self-access), seals (for access monitored by humans)
  - Today: Cards or identity tags (access to buildings), smart cards.
- ***Feature + Bug***
  - Anybody who is in possession of the token has the same rights as the legitimate owner.
  - Physical tokens can be lost or stolen without the user's cooperation
- ***To increase security, physical tokens are often used in combination with something that cannot be stolen***
  - bank cards come with a PIN or with a photo of the user.

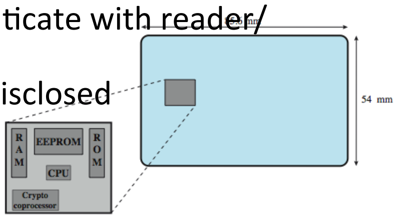


Memory Card

- ***store but do not process data***
- ***magnetic stripe card, e.g. bank card***
- ***electronic memory card***
- ***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

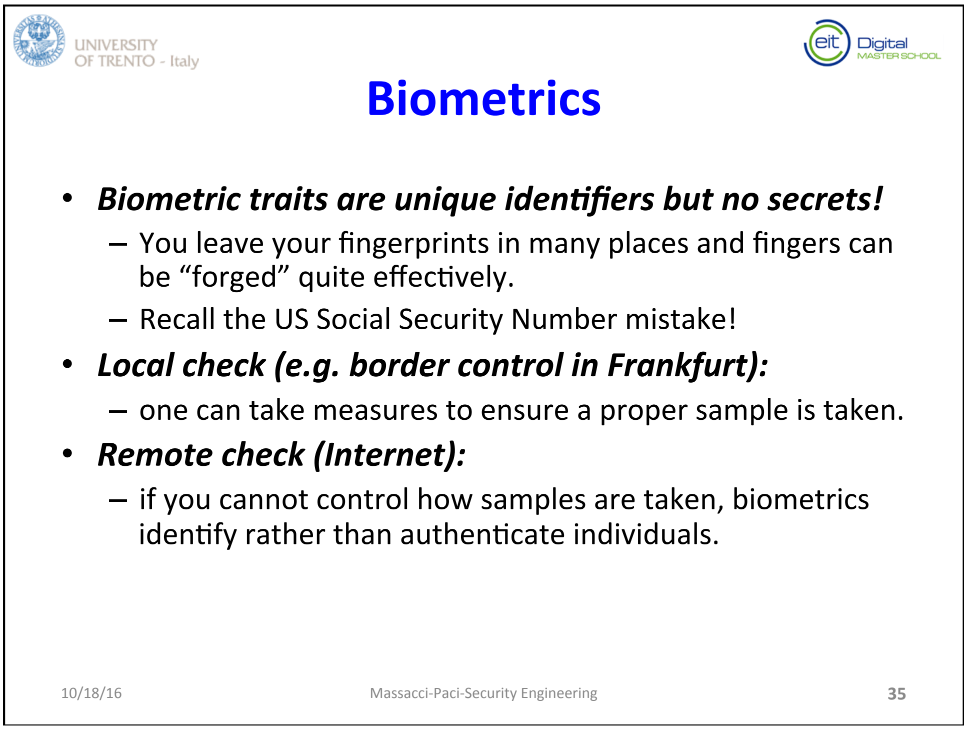
- ***Smartcard has own processor, memory, I/O ports***
  - wired or wireless access by reader
  - may have crypto co-processor
  - ROM, EEPROM, RAM memory
- ***Can store secrets***
  - executes protocol to authenticate with reader/computer
  - secrets are "used" but not disclosed
  - secrets are tamperproof
- ***Alternative: USB dongles***



typical chip layout

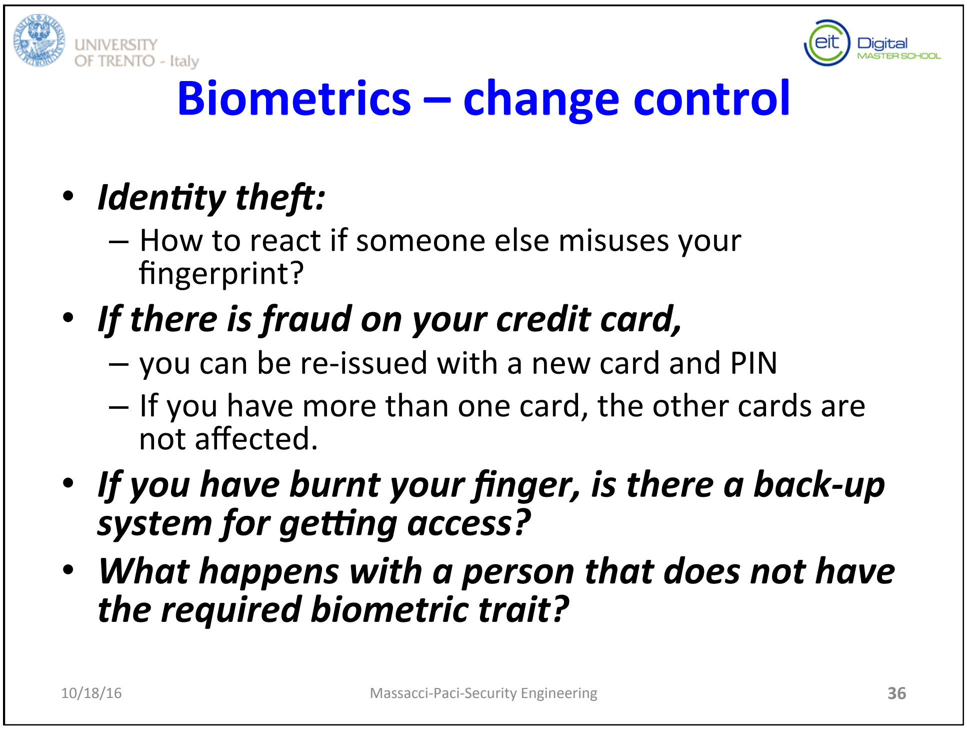
Who You Are

- ***Biometric schemes use unique physical characteristics (traits, features) of a person***
  - face,
  - finger prints,
  - iris patterns,
  - hand geometry
- ***Biometrics may seem to offer the most secure solution for authenticating a person***
  - Very good for specialized/limited access → e.g. access to ACC may require biometric authentication
- ***Little experience from large scale field trials on the performance of biometrics***
  - So far only large scale is biometric on mobile devices, but not know if most people actually turned that on



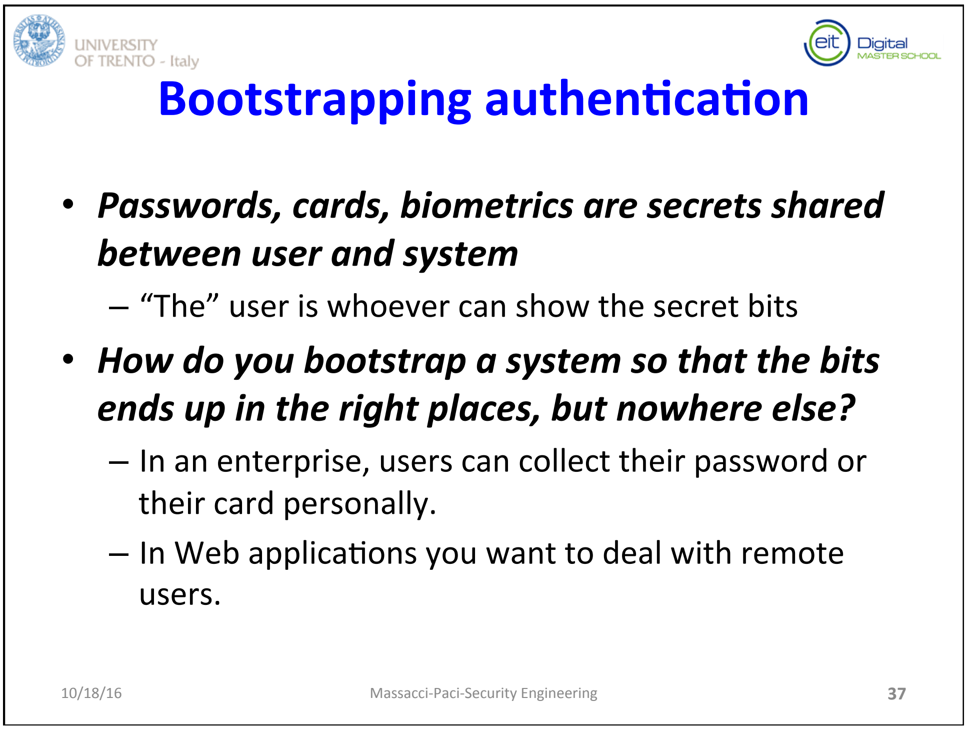
Biometrics

- ***Biometric traits are unique identifiers but no secrets!***
  - You leave your fingerprints in many places and fingers can be “forged” quite effectively.
  - Recall the US Social Security Number mistake!
- ***Local check (e.g. border control in Frankfurt):***
  - one can take measures to ensure a proper sample is taken.
- ***Remote check (Internet):***
  - if you cannot control how samples are taken, biometrics identify rather than authenticate individuals.



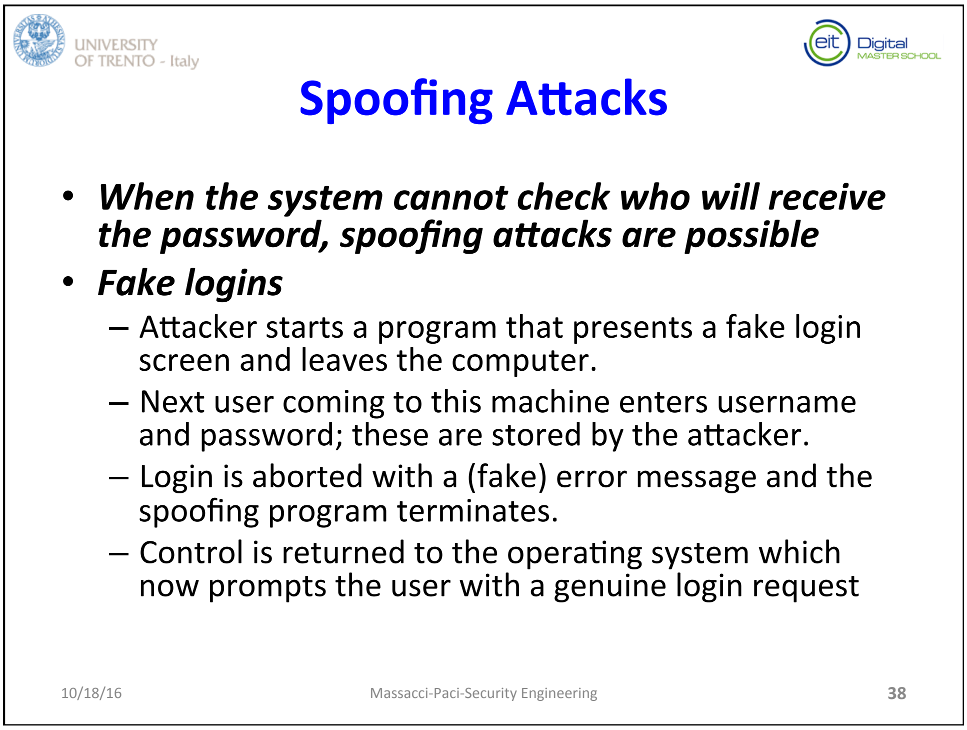
Biometrics – change control

- ***Identity theft:***
  - How to react if someone else misuses your fingerprint?
- ***If there is fraud on your credit card,***
  - you can be re-issued with a new card and PIN
  - If you have more than one card, the other cards are not affected.
- ***If you have burnt your finger, is there a back-up system for getting access?***
- ***What happens with a person that does not have the required biometric trait?***



Bootstrapping authentication

- ***Passwords, cards, biometrics are secrets shared between user and system***
  - "The" user is whoever can show the secret bits
- ***How do you bootstrap a system so that the bits ends up in the right places, but nowhere else?***
  - In an enterprise, users can collect their password or their card personally.
  - In Web applications you want to deal with remote users.



Spoofing Attacks

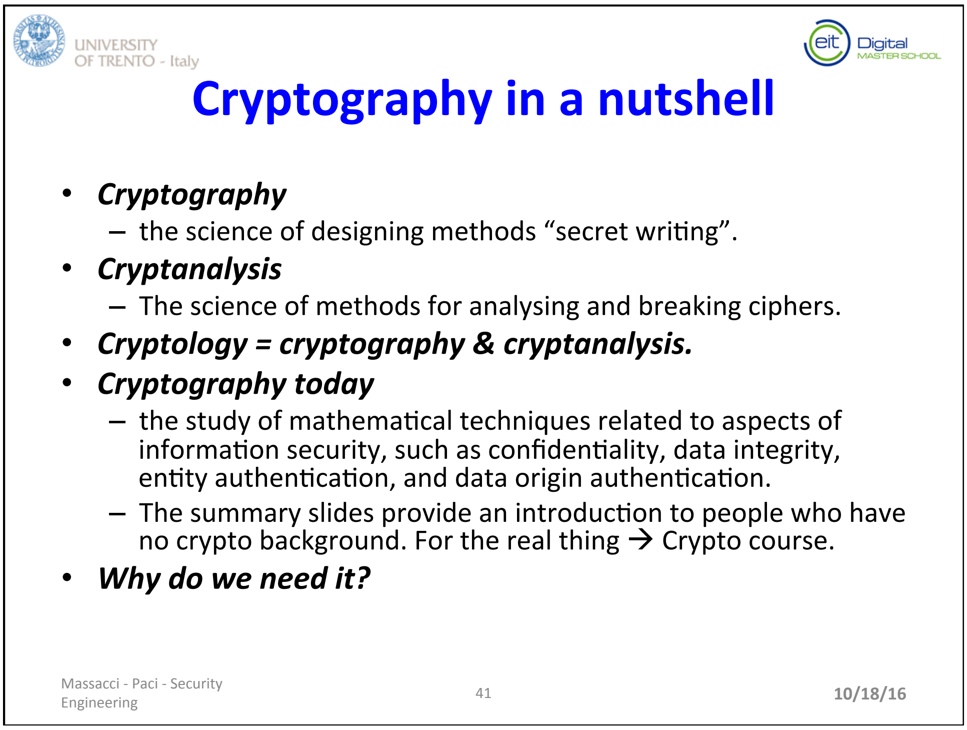
- ***When the system cannot check who will receive the password, spoofing attacks are possible***
- ***Fake logins***
  - Attacker starts a program that presents a fake login screen and leaves the computer.
  - Next user coming to this machine enters username and password; these are stored by the attacker.
  - Login is aborted with a (fake) error message and the spoofing program terminates.
  - Control is returned to the operating system which now prompts the user with a genuine login request

Authention of a Remote User

- ***Send passwords that are valid only for a single log-in request so that the user has to change immediately to a password not known by the sender***
  - Assume attacker does not control server's email, backbone network, local network, local email
- ***Request confirmation on a different channel to activate user account,***
  - Enter password on a webpage and send confirmation by SMS.
  - Send mail by courier with personal delivery.
- ***In an organisation:***
  - Don't give password to caller but call back an authorized phone number, e.g. from an internal company address book.
  - Call back someone else, e.g. caller's manager or local security officer
- ***Just multiple path not enough protection against insider attacks***

Is this just theory?

- ***Traditional Credit Card Scam by Bank Employees***
  - Change card adress to insider's address on database
  - Re-issue PIN via email and intercept PIN
  - Switch back address to customer original address
  - Happily spend money on behalf of customer
- ***Zeus – "Man in the Browser" attack on e-banking authentication system***
  - Bank requires
    - User password to log in on the system
    - one time password to make a bank transfer
  - How Zeus managed to bypass that?
  - Which solutions the bank devised?

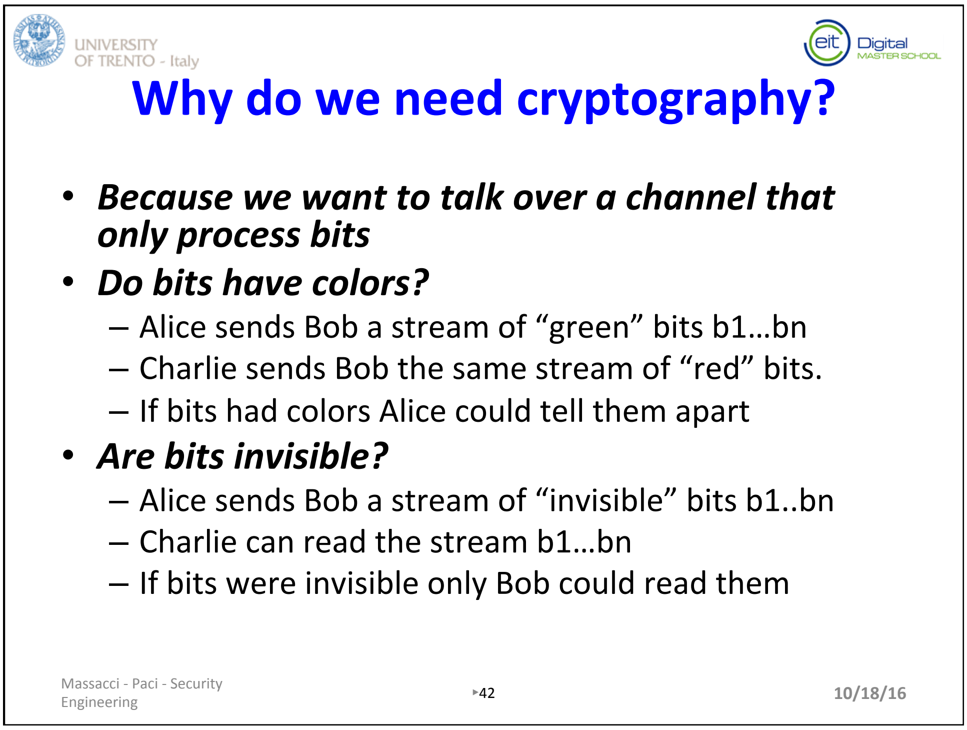


UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Cryptography in a nutshell

- ***Cryptography***
  - the science of designing methods “secret writing”.
- ***Cryptanalysis***
  - The science of methods for analysing and breaking ciphers.
- ***Cryptology = cryptography & cryptanalysis.***
- ***Cryptography today***
  - the study of mathematical techniques related to aspects of information security, such as confidentiality, data integrity, entity authentication, and data origin authentication.
  - The summary slides provide an introduction to people who have no crypto background. For the real thing → Crypto course.
- ***Why do we need it?***

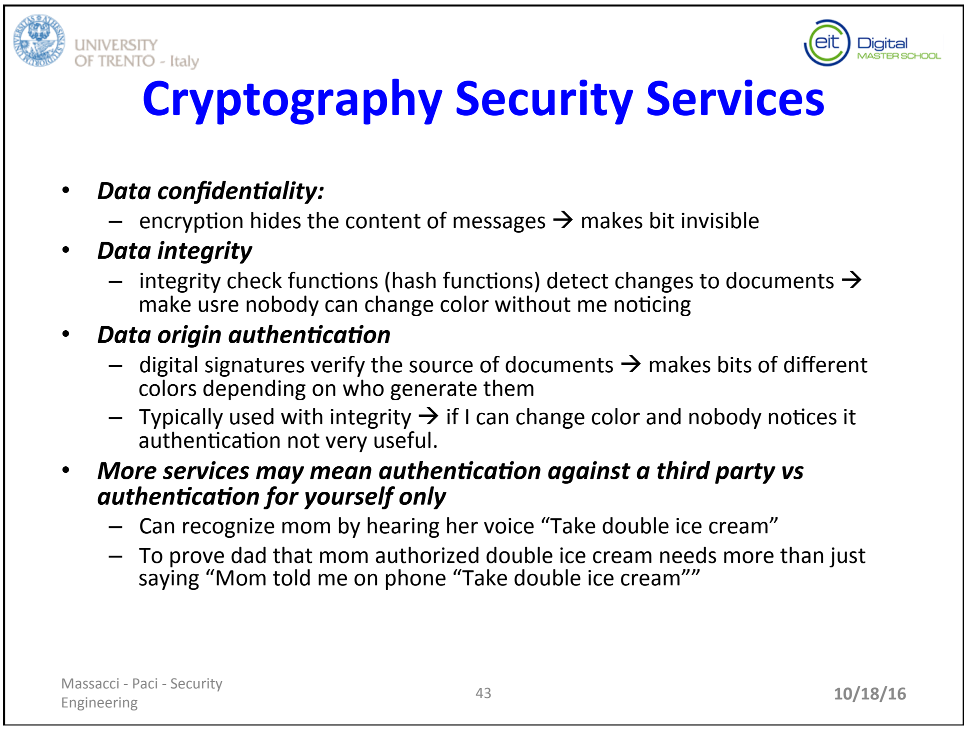


UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

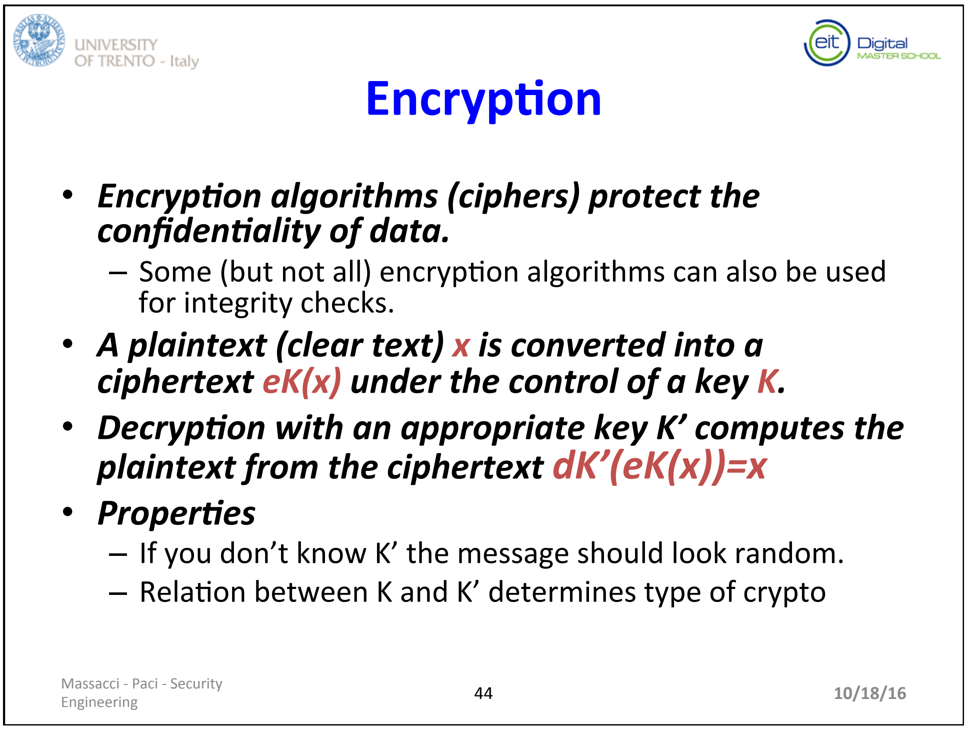
Why do we need cryptography?

- ***Because we want to talk over a channel that only process bits***
- ***Do bits have colors?***
  - Alice sends Bob a stream of “green” bits b1…bn
  - Charlie sends Bob the same stream of “red” bits.
  - If bits had colors Alice could tell them apart
- ***Are bits invisible?***
  - Alice sends Bob a stream of “invisible” bits b1..bn
  - Charlie can read the stream b1…bn
  - If bits were invisible only Bob could read them



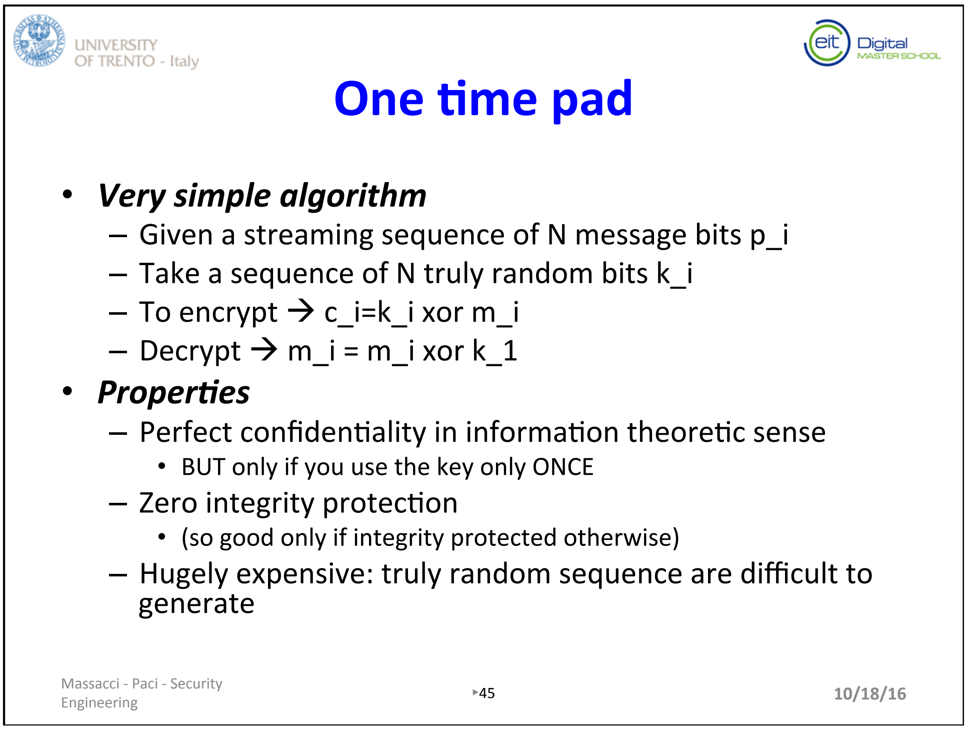
Cryptography Security Services

- ***Data confidentiality:***
  - encryption hides the content of messages → makes bit invisible
- ***Data integrity***
  - integrity check functions (hash functions) detect changes to documents → make usre nobody can change color without me noticing
- ***Data origin authentication***
  - digital signatures verify the source of documents → makes bits of different colors depending on who generate them
  - Typically used with integrity → if I can change color and nobody notices it authentication not very useful.
- ***More services may mean authentication against a third party vs authentication for yourself only***
  - Can recognize mom by hearing her voice "Take double ice cream"
  - To prove dad that mom authorized double ice cream needs more than just saying "Mom told me on phone "Take double ice cream""



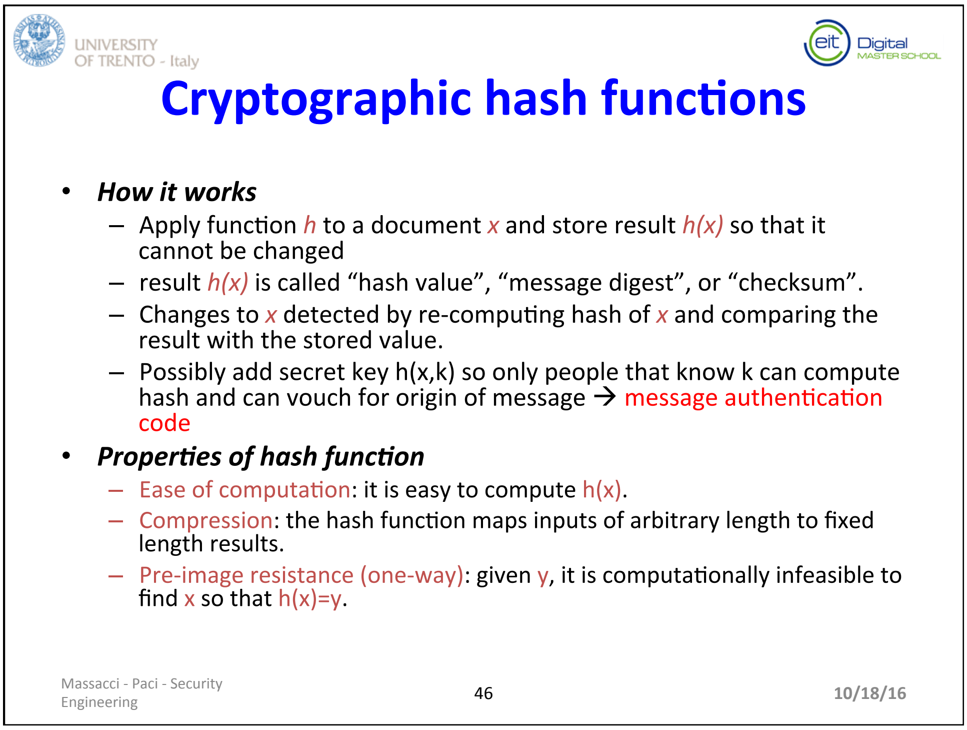
Encryption

- ***Encryption algorithms (ciphers) protect the confidentiality of data.***
  - Some (but not all) encryption algorithms can also be used for integrity checks.
- ***A plaintext (clear text) x is converted into a ciphertext eK(x) under the control of a key K.***
- ***Decryption with an appropriate key K' computes the plaintext from the ciphertext dK'(eK(x))=x***
- ***Properties***
  - If you don't know K' the message should look random.
  - Relation between K and K' determines type of crypto



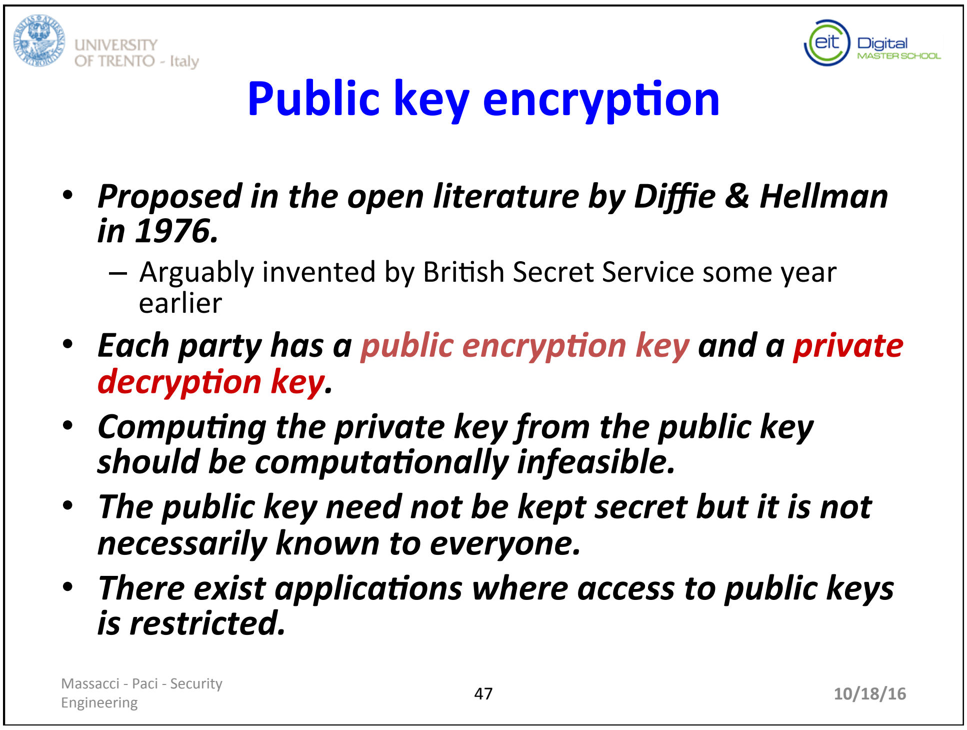
One time pad

- ***Very simple algorithm***
  - Given a streaming sequence of N message bits p_i
  - Take a sequence of N truly random bits k_i
  - To encrypt → c_i=k_i xor m_i
  - Decrypt → m_i = m_i xor k_1
- ***Properties***
  - Perfect confidentiality in information theoretic sense
    - BUT only if you use the key only ONCE
  - Zero integrity protection
    - (so good only if integrity protected otherwise)
  - Hugely expensive: truly random sequence are difficult to generate



Cryptographic hash functions

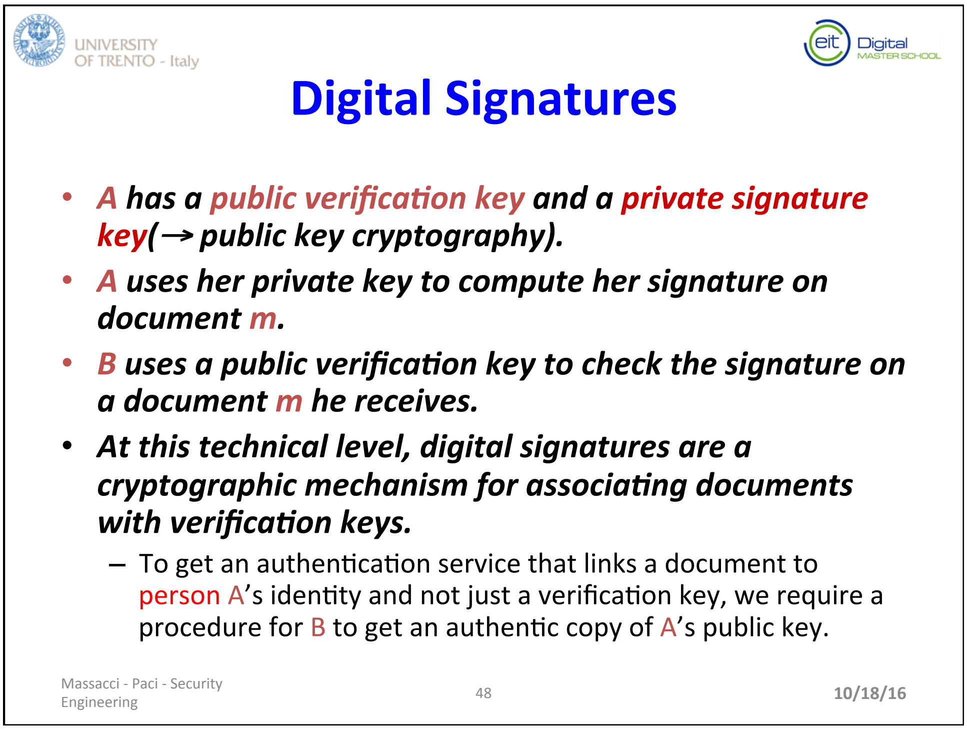
- ***How it works***
  - Apply function *h* to a document *x* and store result *h(x)* so that it cannot be changed
  - result *h(x)* is called “hash value”, “message digest”, or “checksum”.
  - Changes to *x* detected by re-computing hash of *x* and comparing the result with the stored value.
  - Possibly add secret key h(x,k) so only people that know k can compute hash and can vouch for origin of message → message authentication code
- ***Properties of hash function***
  - Ease of computation: it is easy to compute h(x).
  - Compression: the hash function maps inputs of arbitrary length to fixed length results.
  - Pre-image resistance (one-way): given y, it is computationally infeasible to find x so that h(x)=y.



Public key encryption

- ***Proposed in the open literature by Diffie & Hellman in 1976.***
  - Arguably invented by British Secret Service some year earlier
- ***Each party has a public encryption key and a private decryption key.***
- ***Computing the private key from the public key should be computationally infeasible.***
- ***The public key need not be kept secret but it is not necessarily known to everyone.***
- ***There exist applications where access to public keys is restricted.***

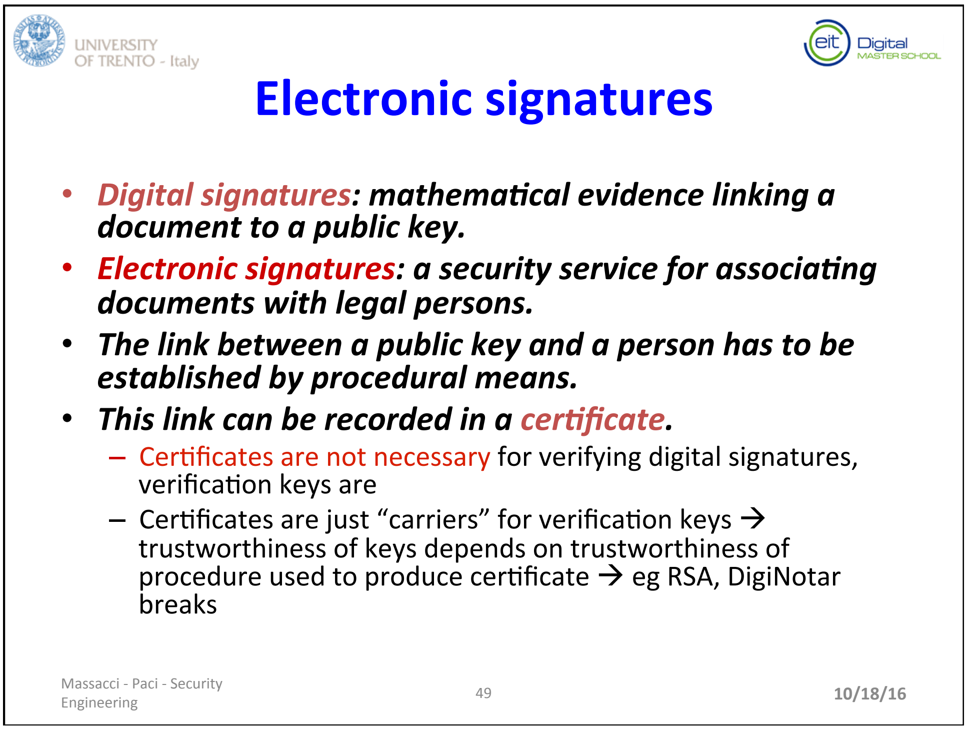
Massacci - Paci - Security Engineering 47 10/18/16



Digital Signatures

- ***A has a public verification key and a private signature key(→ public key cryptography).***
- ***A uses her private key to compute her signature on document m.***
- ***B uses a public verification key to check the signature on a document m he receives.***
- ***At this technical level, digital signatures are a cryptographic mechanism for associating documents with verification keys.***
  - To get an authentication service that links a document to person A's identity and not just a verification key, we require a procedure for B to get an authentic copy of A's public key.

Massacci - Paci - Security Engineering 48 10/18/16

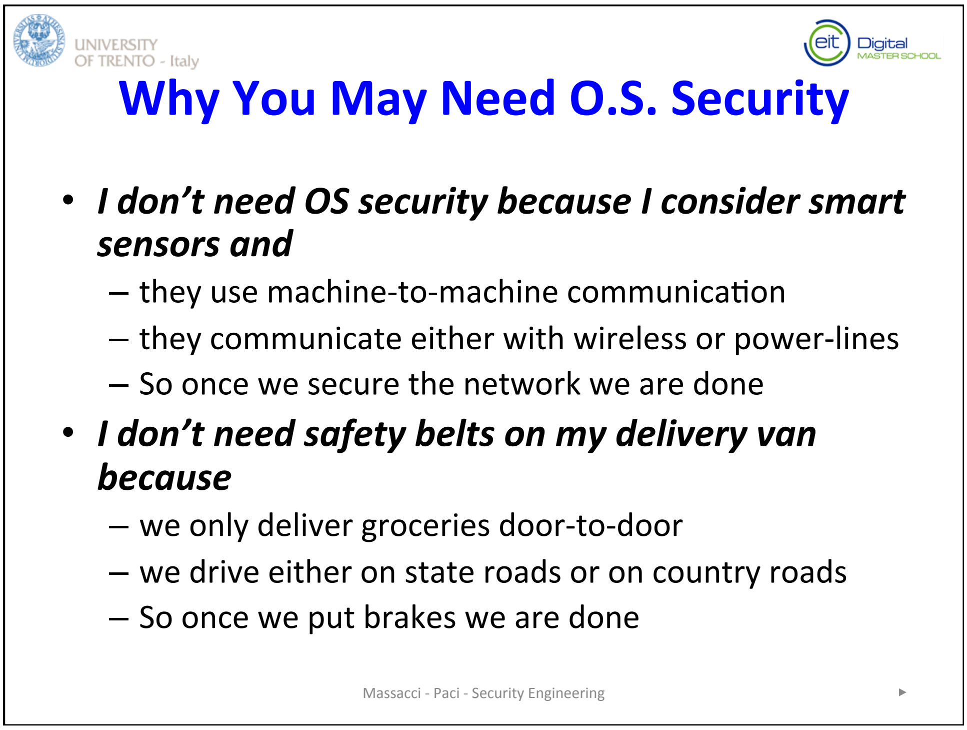


UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Electronic signatures

- ***Digital signatures*: mathematical evidence linking a document to a public key.**
- ***Electronic signatures*: a security service for associating documents with legal persons.**
- ***The link between a public key and a person has to be established by procedural means.***
- ***This link can be recorded in a certificate.***
  - Certificates are not necessary for verifying digital signatures, verification keys are
  - Certificates are just "carriers" for verification keys → trustworthiness of keys depends on trustworthiness of procedure used to produce certificate → eg RSA, DigiNotar breaks



UNIVERSITY OF TRENTO - Italy

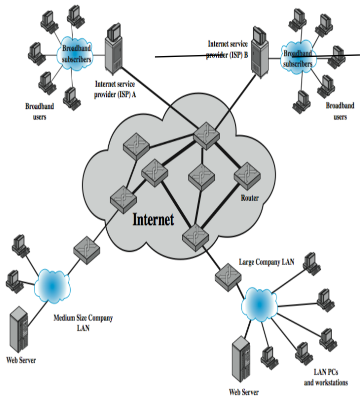
eit Digital MASTER SCHOOL

Why You May Need O.S. Security

- ***I don't need OS security because I consider smart sensors and***
  - they use machine-to-machine communication
  - they communicate either with wireless or power-lines
  - So once we secure the network we are done
- ***I don't need safety belts on my delivery van because***
  - we only deliver groceries door-to-door
  - we drive either on state roads or on country roads
  - So once we put brakes we are done

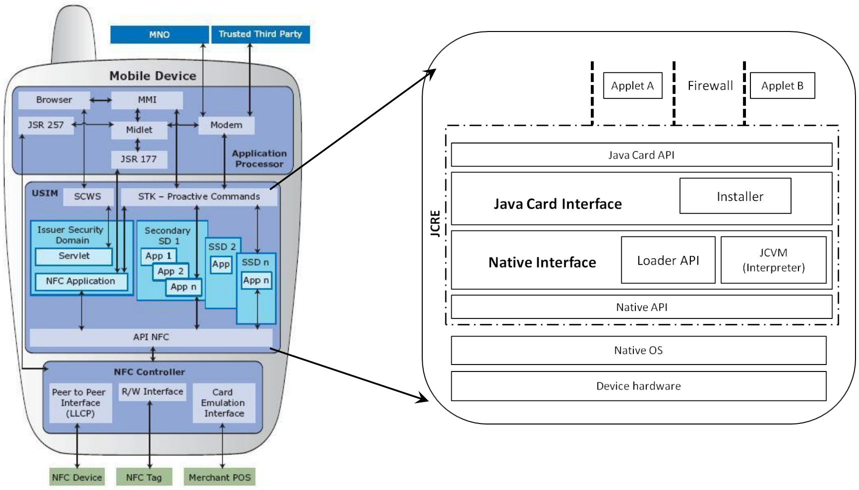
Some Misinterpreted Pictures..

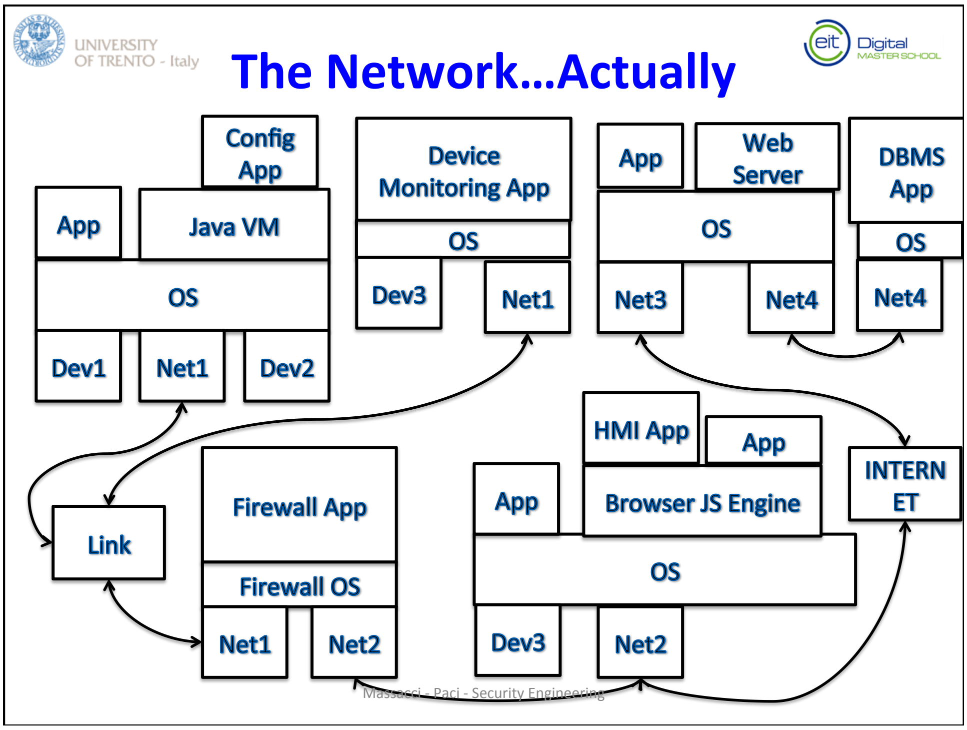
- ***The picture is "evocative"***
  - but this is NOT the reality
- ***A "descriptive" picture would include all the different software and protocol stacks***
  - A MSc student in CS should know the actual reality…
  - And reason on what is really going on



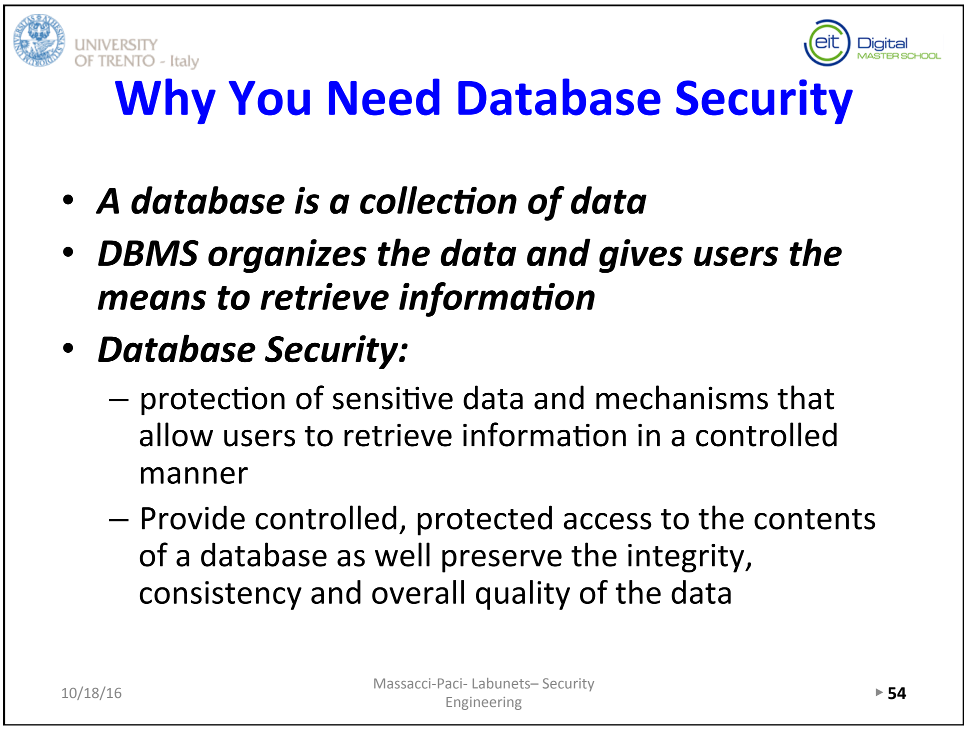
What is a smart sensor?

- ***Basically a Phone with a GSM Card***



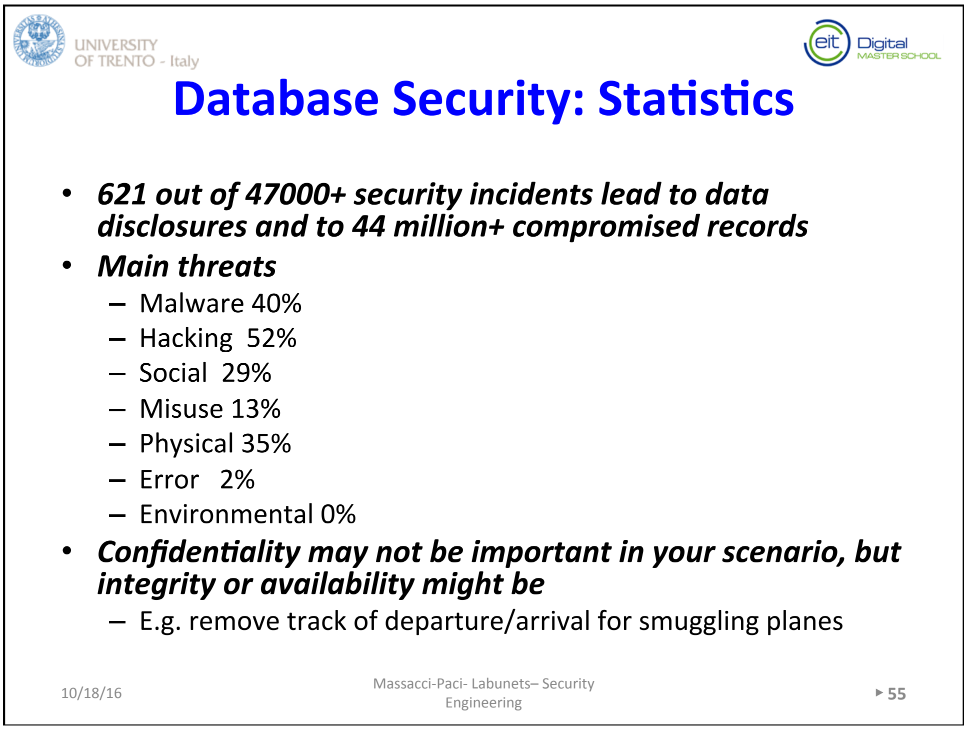


The Network…Actually



Why You Need Database Security

- ***A database is a collection of data***
- ***DBMS organizes the data and gives users the means to retrieve information***
- ***Database Security:***
  - protection of sensitive data and mechanisms that allow users to retrieve information in a controlled manner
  - Provide controlled, protected access to the contents of a database as well preserve the integrity, consistency and overall quality of the data

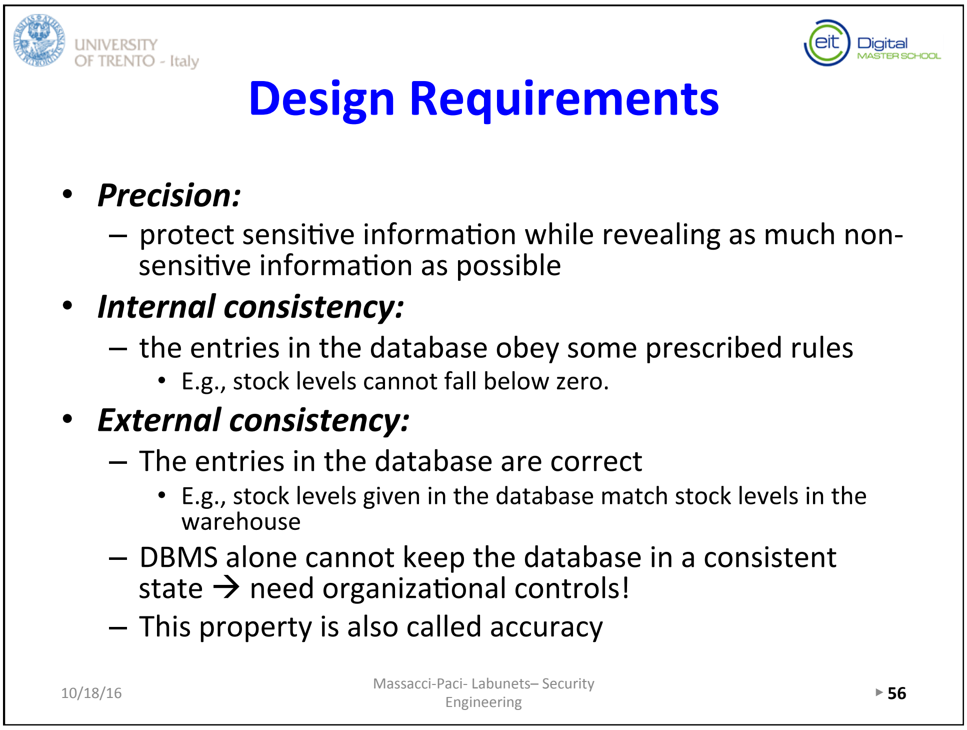


UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Database Security: Statistics

- ***621 out of 47000+ security incidents lead to data disclosures and to 44 million+ compromised records***
- ***Main threats***
  - Malware 40%
  - Hacking 52%
  - Social 29%
  - Misuse 13%
  - Physical 35%
  - Error 2%
  - Environmental 0%
- ***Confidentiality may not be important in your scenario, but integrity or availability might be***
  - E.g. remove track of departure/arrival for smuggling planes

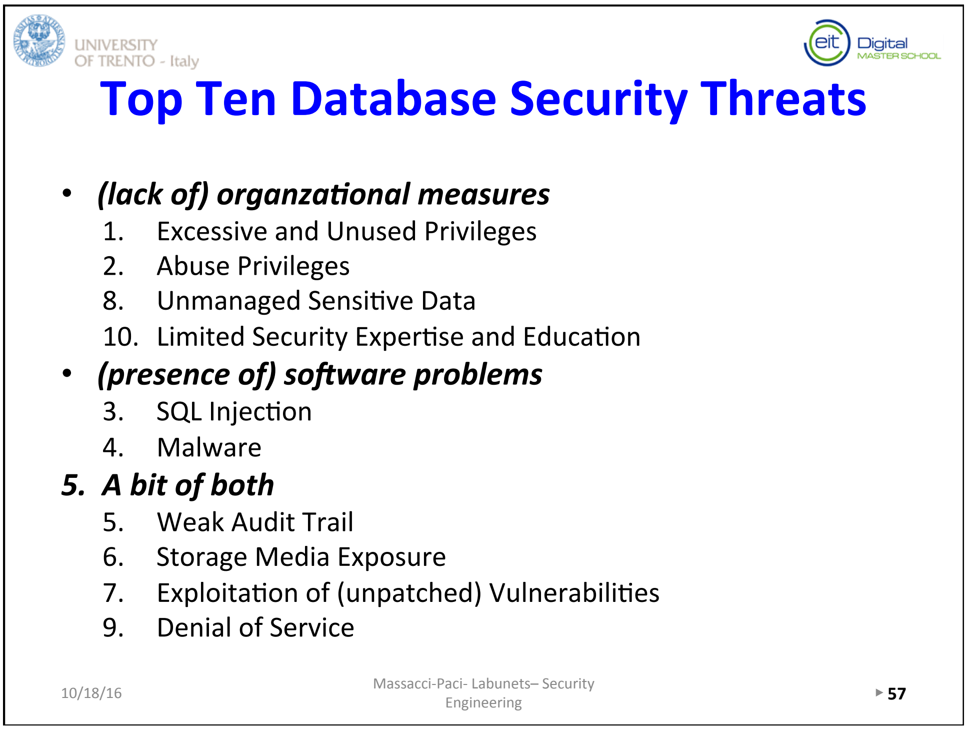


UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Design Requirements

- ***Precision:***
  - protect sensitive information while revealing as much non-sensitive information as possible
- ***Internal consistency:***
  - the entries in the database obey some prescribed rules
    - E.g., stock levels cannot fall below zero.
- ***External consistency:***
  - The entries in the database are correct
    - E.g., stock levels given in the database match stock levels in the warehouse
  - DBMS alone cannot keep the database in a consistent state → need organizational controls!
  - This property is also called accuracy

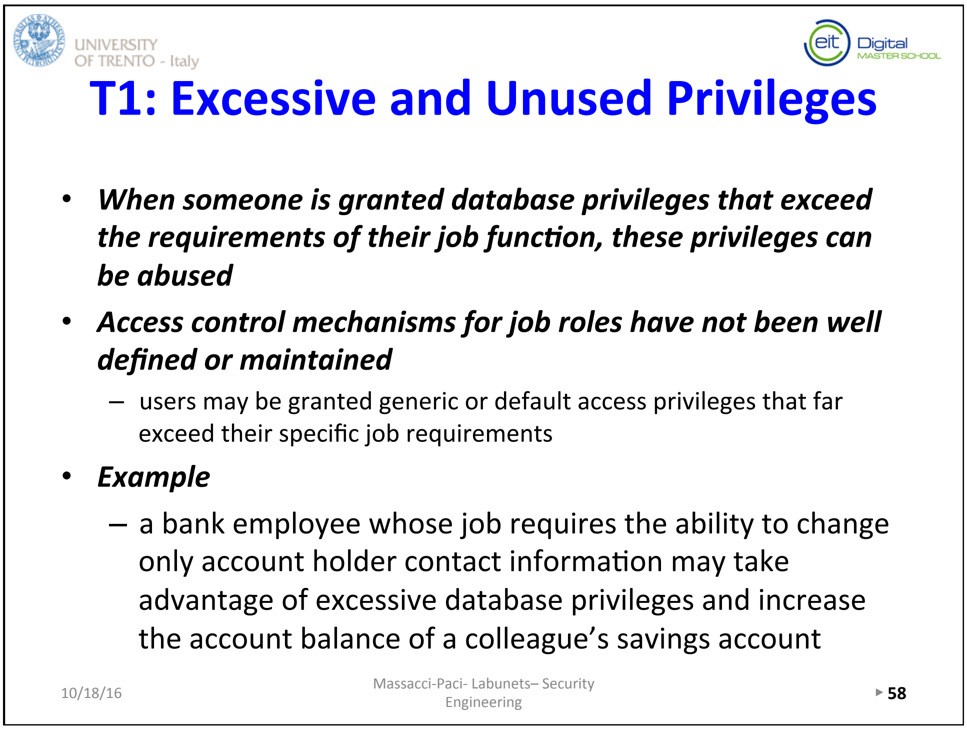


Top Ten Database Security Threats

- ***(lack of) organzational measures***
  1. Excessive and Unused Privileges
  2. Abuse Privileges
  8. Unmanaged Sensitive Data
  10. Limited Security Expertise and Education
- ***(presence of) software problems***
  3. SQL Injection
  4. Malware

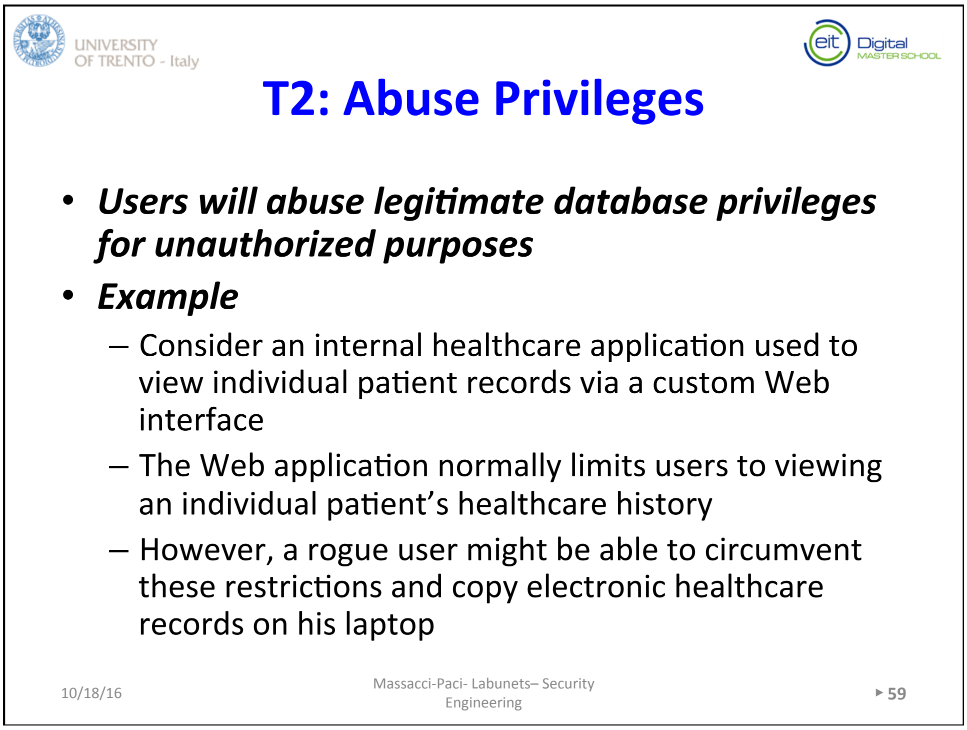
5. A bit of both

  5. Weak Audit Trail
  6. Storage Media Exposure
  7. Exploitation of (unpatched) Vulnerabilities
  9. Denial of Service



T1: Excessive and Unused Privileges

- ***When someone is granted database privileges that exceed the requirements of their job function, these privileges can be abused***
- ***Access control mechanisms for job roles have not been well defined or maintained***
  – users may be granted generic or default access privileges that far exceed their specific job requirements
- ***Example***
  – a bank employee whose job requires the ability to change only account holder contact information may take advantage of excessive database privileges and increase the account balance of a colleague’s savings account

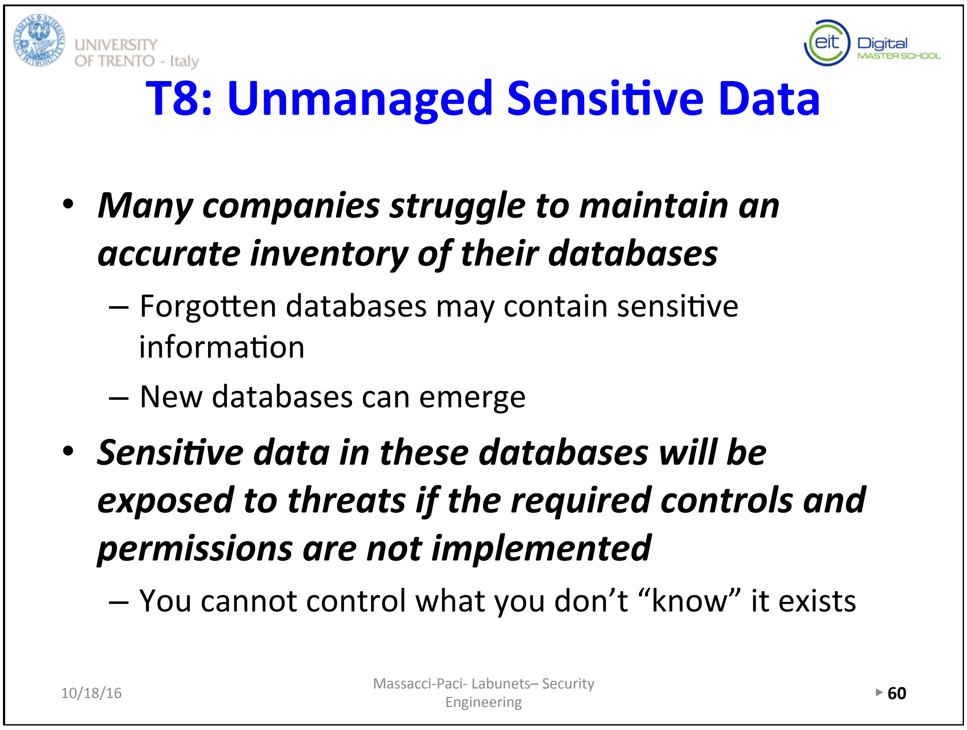


UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

T2: Abuse Privileges

- ***Users will abuse legitimate database privileges for unauthorized purposes***
- ***Example***
  - Consider an internal healthcare application used to view individual patient records via a custom Web interface
  - The Web application normally limits users to viewing an individual patient's healthcare history
  - However, a rogue user might be able to circumvent these restrictions and copy electronic healthcare records on his laptop



UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

T8: Unmanaged Sensitive Data

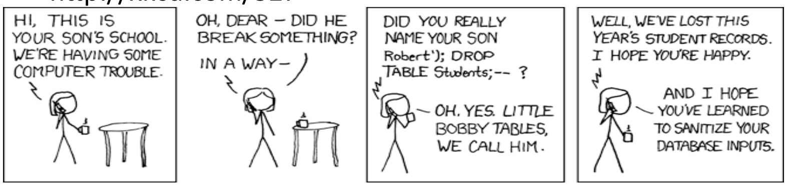
- ***Many companies struggle to maintain an accurate inventory of their databases***
  - Forgotten databases may contain sensitive information
  - New databases can emerge
- ***Sensitive data in these databases will be exposed to threats if the required controls and permissions are not implemented***
  - You cannot control what you don't "know" it exists

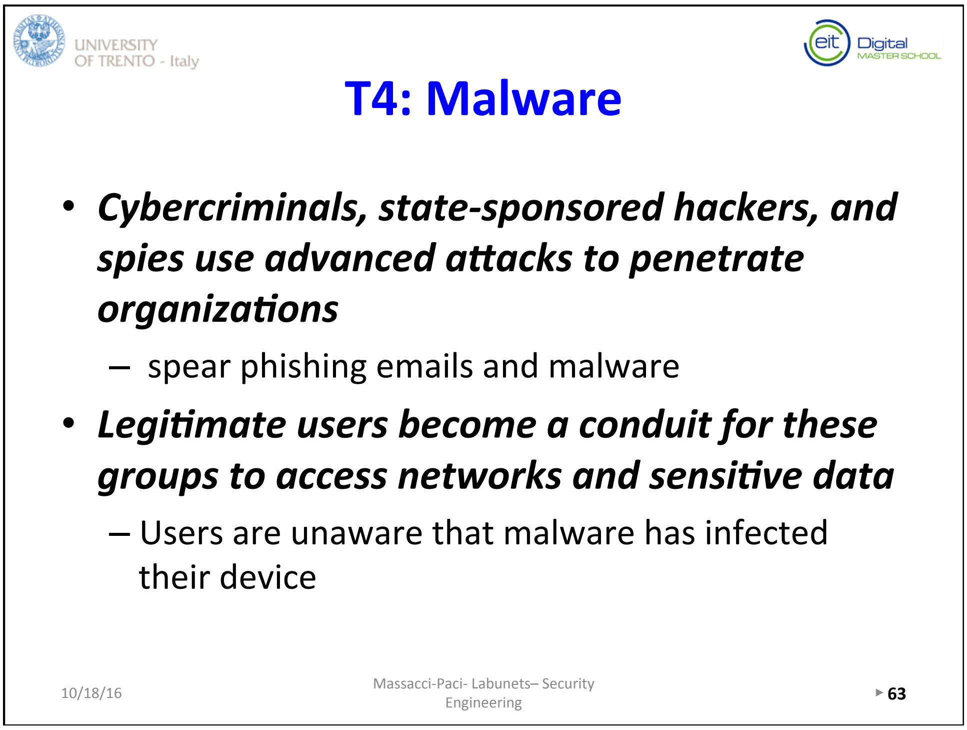
T10: Limited Security Expertise and Education

- ***Internal security controls are not keeping pace with data growth and many organizations are ill-equipped to deal with a security breach***
- ***Due to the lack of expertise required to implement security controls, policies, and training***
- ***According to PWC's 2012 Information Security Breaches Survey***
  - 75% of the organizations surveyed experienced staff-related breaches when a security policy was poorly understood
  - 54% of small businesses did not have a program for educating their staff about security risks

T3: SQL Injection

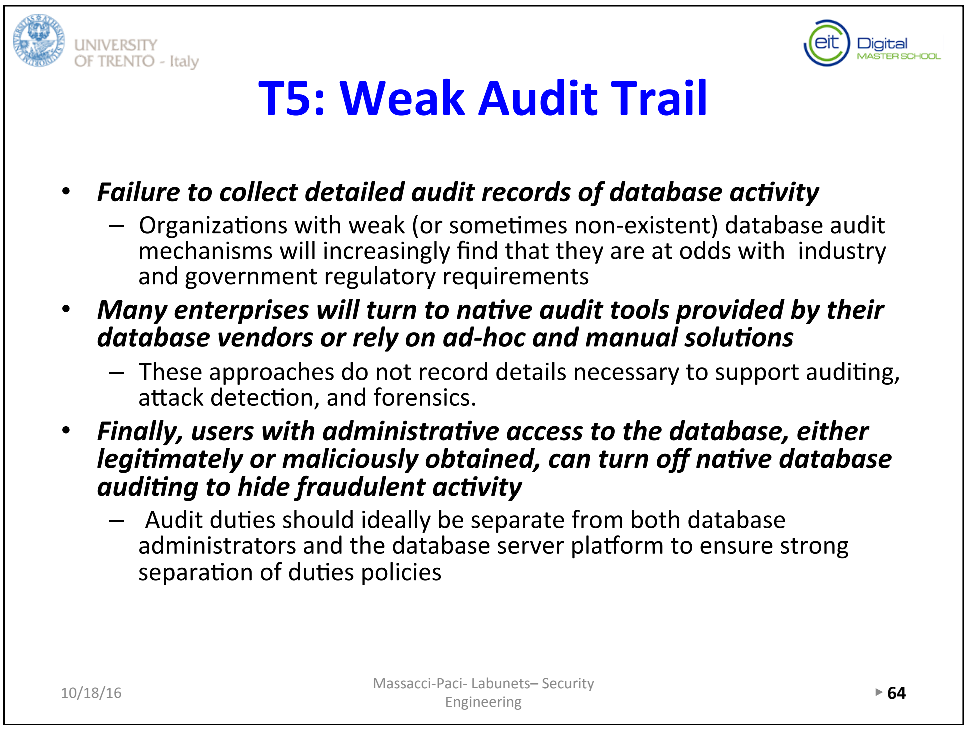
- ***Already discussed for application security***
  - SQL injection involves inserting (or "injecting") unauthorized or malicious database statements into a vulnerable SQL data channel such as a Web application or stored procedure
  - If these injected statements are executed by the database, critical data stores can be viewed, copied, and altered
- ***Normally this happens by exploiting the polyglottism of the web application on top of the DB***
  - http://xkcd.com/327


HI, THIS IS YOUR SON'S SCHOOL. WE'RE HAVING SOME COMPUTER TROUBLE.
OH, DEAR – DID HE BREAK SOMETHING?
IN A WAY–
DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?
OH. YES. LITTLE BOBBY TABLES, WE CALL HIM.
WELL, WE'VE LOST THIS YEAR'S STUDENT RECORDS. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.



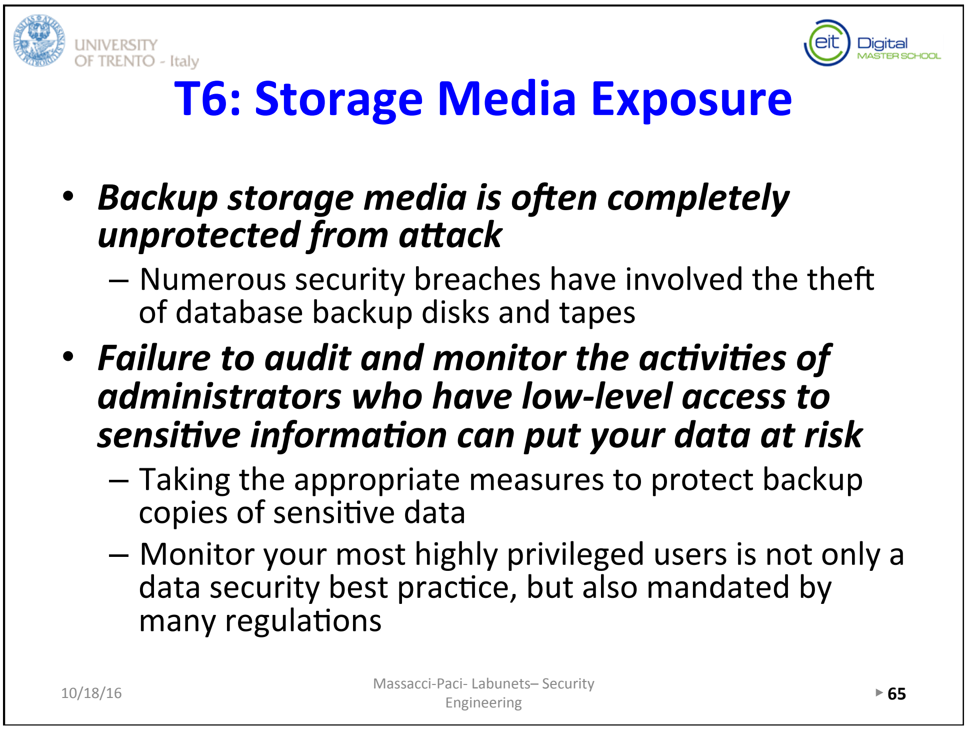
T4: Malware

- ***Cybercriminals, state-sponsored hackers, and spies use advanced attacks to penetrate organizations***
  - spear phishing emails and malware
- ***Legitimate users become a conduit for these groups to access networks and sensitive data***
  - Users are unaware that malware has infected their device



T5: Weak Audit Trail

- ***Failure to collect detailed audit records of database activity***
  - Organizations with weak (or sometimes non-existent) database audit mechanisms will increasingly find that they are at odds with industry and government regulatory requirements
- ***Many enterprises will turn to native audit tools provided by their database vendors or rely on ad-hoc and manual solutions***
  - These approaches do not record details necessary to support auditing, attack detection, and forensics.
- ***Finally, users with administrative access to the database, either legitimately or maliciously obtained, can turn off native database auditing to hide fraudulent activity***
  - Audit duties should ideally be separate from both database administrators and the database server platform to ensure strong separation of duties policies

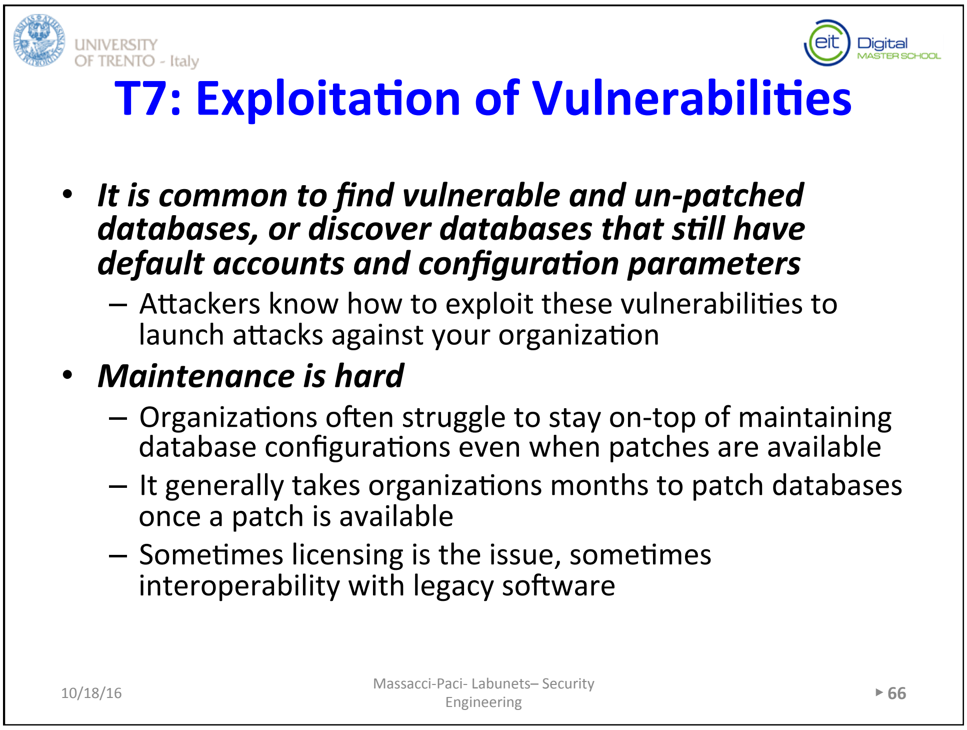


UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

T6: Storage Media Exposure

- ***Backup storage media is often completely unprotected from attack***
  - Numerous security breaches have involved the theft of database backup disks and tapes
- ***Failure to audit and monitor the activities of administrators who have low-level access to sensitive information can put your data at risk***
  - Taking the appropriate measures to protect backup copies of sensitive data
  - Monitor your most highly privileged users is not only a data security best practice, but also mandated by many regulations

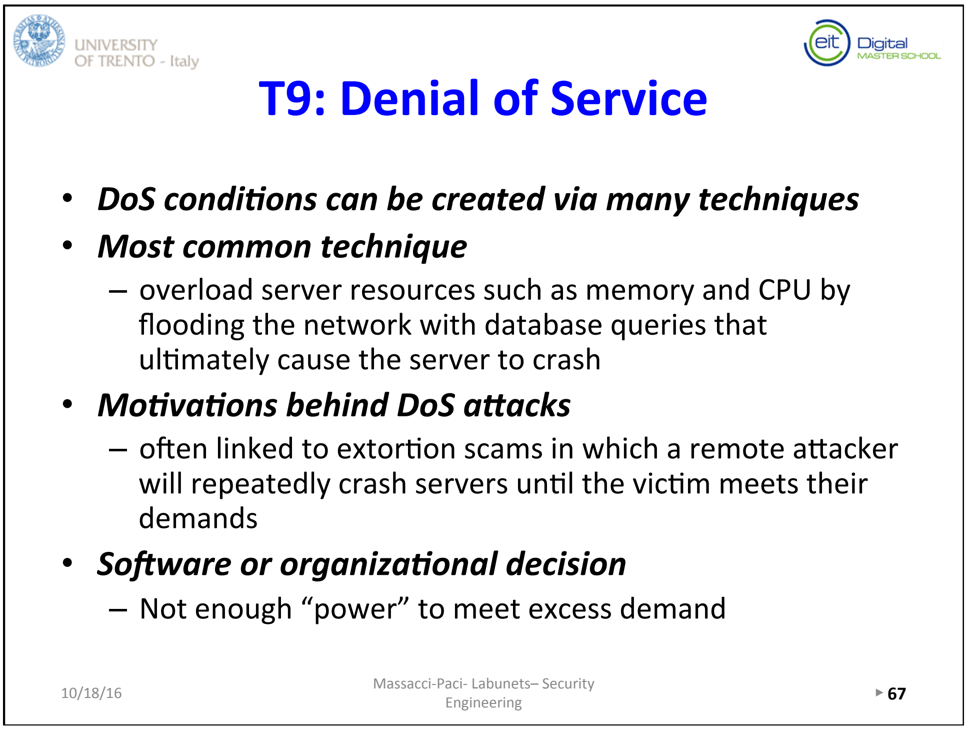


UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

T7: Exploitation of Vulnerabilities

- ***It is common to find vulnerable and un-patched databases, or discover databases that still have default accounts and configuration parameters***
  - Attackers know how to exploit these vulnerabilities to launch attacks against your organization
- ***Maintenance is hard***
  - Organizations often struggle to stay on-top of maintaining database configurations even when patches are available
  - It generally takes organizations months to patch databases once a patch is available
  - Sometimes licensing is the issue, sometimes interoperability with legacy software



UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

T9: Denial of Service

- ***DoS conditions can be created via many techniques***
- ***Most common technique***
  - overload server resources such as memory and CPU by flooding the network with database queries that ultimately cause the server to crash
- ***Motivations behind DoS attacks***
  - often linked to extortion scams in which a remote attacker will repeatedly crash servers until the victim meets their demands
- ***Software or organizational decision***
  - Not enough "power" to meet excess demand