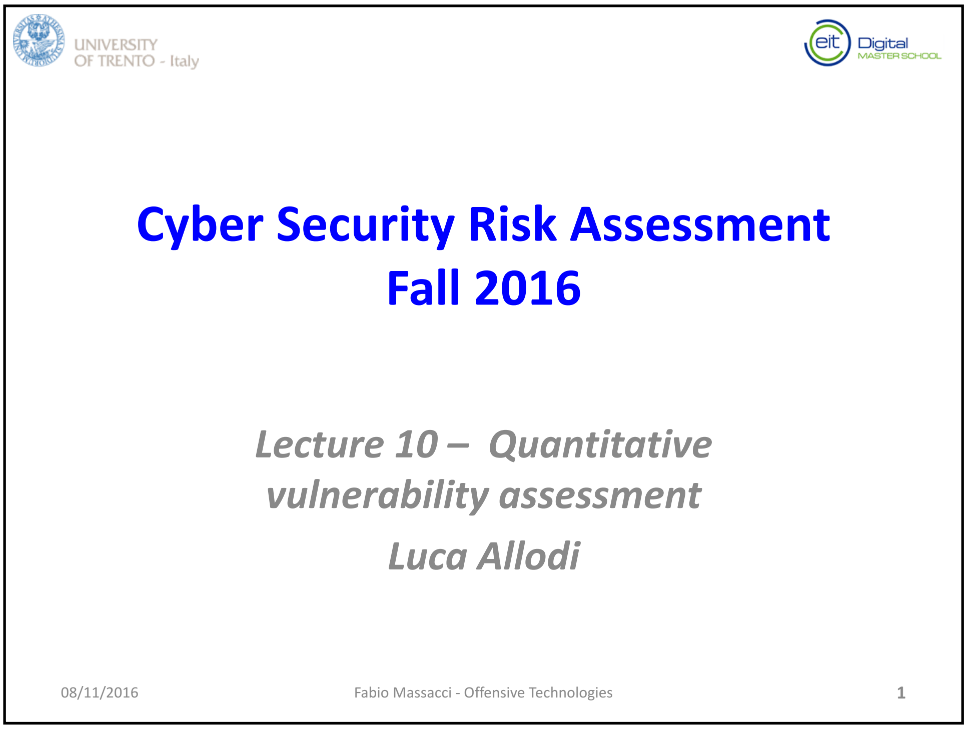
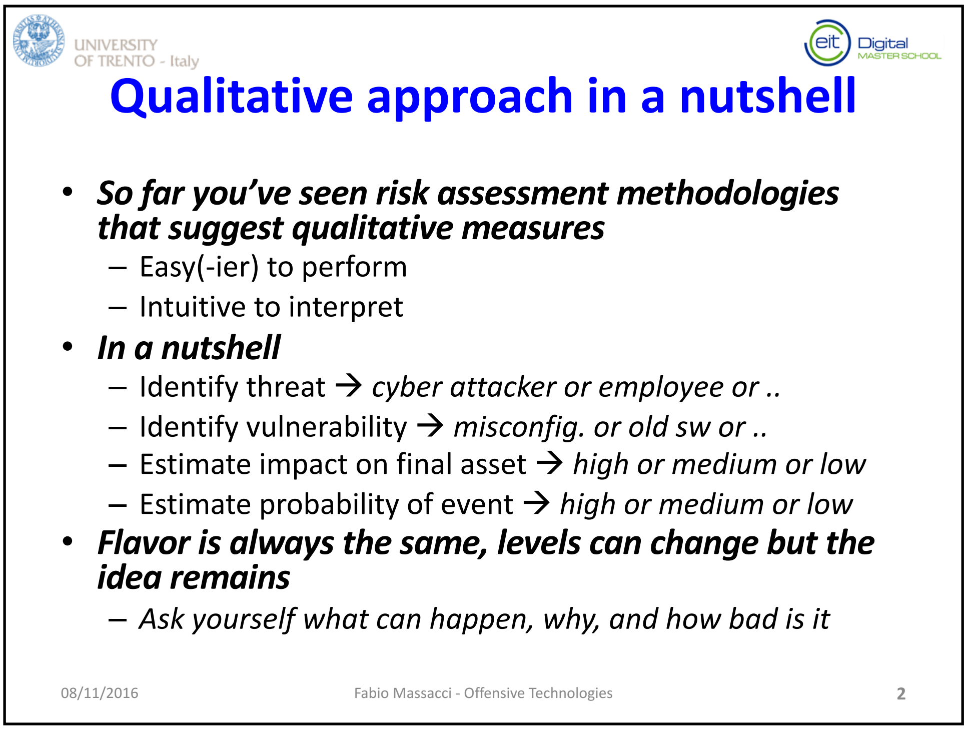


Cyber Security Risk Assessment Fall 2016

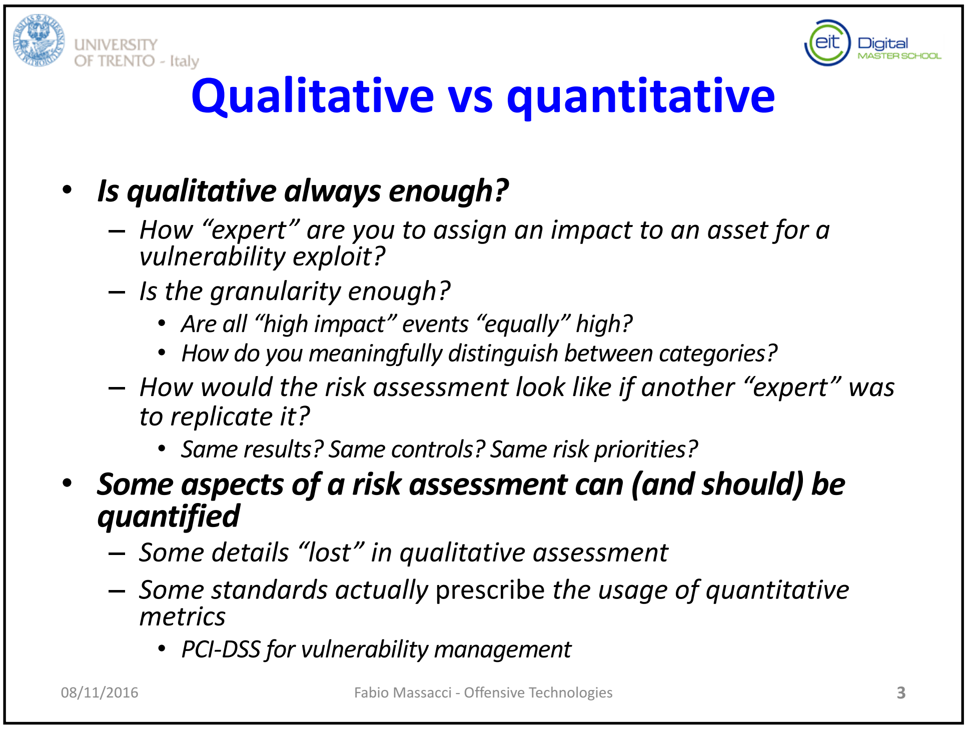
Lecture 10 – Quantitative vulnerability assessment

Luca Allodi



Qualitative approach in a nutshell

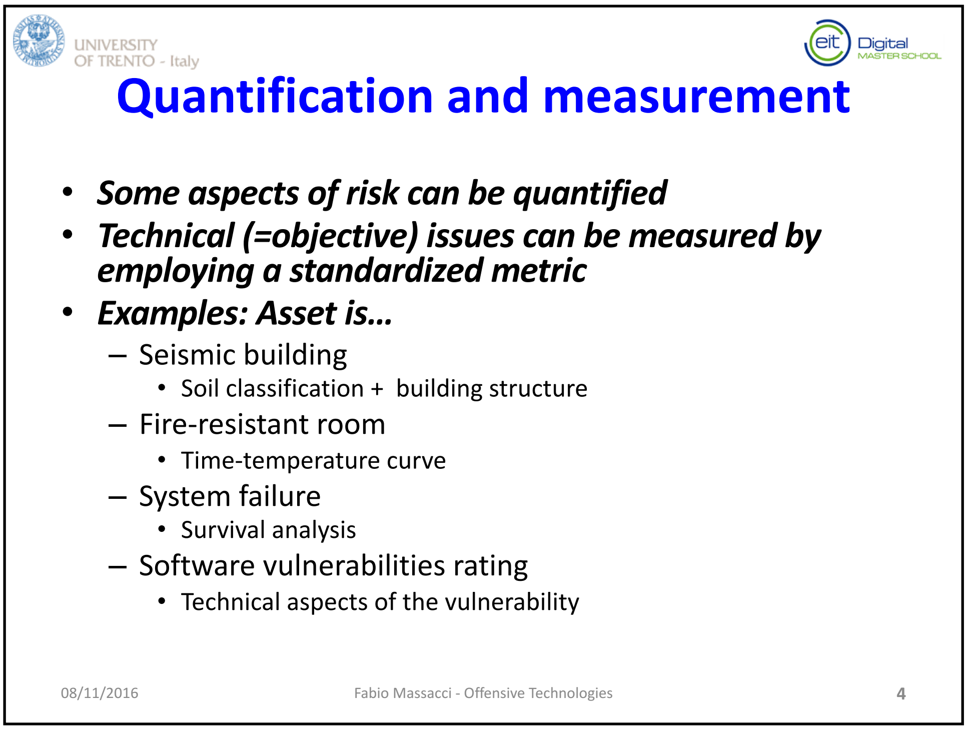
- ***So far you've seen risk assessment methodologies that suggest qualitative measures***
  - Easy(-ier) to perform
  - Intuitive to interpret
- ***In a nutshell***
  - Identify threat → *cyber attacker or employee or ..*
  - Identify vulnerability → *misconfig. or old sw or ..*
  - Estimate impact on final asset → *high or medium or low*
  - Estimate probability of event → *high or medium or low*
- ***Flavor is always the same, levels can change but the idea remains***
  - *Ask yourself what can happen, why, and how bad is it*



Qualitative vs quantitative

- ***Is qualitative always enough?***
  - *How "expert" are you to assign an impact to an asset for a vulnerability exploit?*
  - *Is the granularity enough?*
    - *Are all "high impact" events "equally" high?*
    - *How do you meaningfully distinguish between categories?*
  - *How would the risk assessment look like if another "expert" was to replicate it?*
    - *Same results? Same controls? Same risk priorities?*
- ***Some aspects of a risk assessment can (and should) be quantified***
  - *Some details "lost" in qualitative assessment*
  - *Some standards actually* prescribe *the usage of quantitative metrics*
    - *PCI-DSS for vulnerability management*

08/11/2016 Fabio Massacci - Offensive Technologies 3



Quantification and measurement

- ***Some aspects of risk can be quantified***
- ***Technical (=objective) issues can be measured by employing a standardized metric***
- ***Examples: Asset is…***
  - Seismic building
    - Soil classification + building structure
  - Fire-resistant room
    - Time-temperature curve
  - System failure
    - Survival analysis
  - Software vulnerabilities rating
    - Technical aspects of the vulnerability

08/11/2016 Fabio Massacci - Offensive Technologies 4

UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Example of qualitative vs quantitative

From lecture 05, slide 25

Threat Source	Threat Event	Impact
Alice	Install Malware	Moderate
Outsider	SQL Injection	High

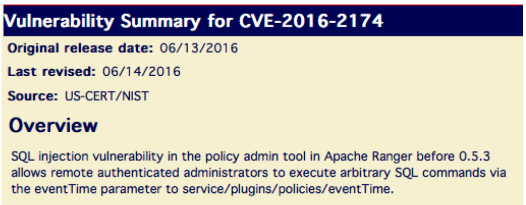
- ***Qualitative assessment***
  - Malware has a lower impact than SQLi → assigned based on expert judgment
- ***Result:***
  - First fix SQL injection because it has a high impact
    - Confidentiality and Integrity impacts on data
  - Then add controls for malware (update AV, data caps policies,..)
    - Worrisome but moderated impact
    - Disclosure of only some data/compartmentalization

UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Example of qualitative vs quantitative

- ***Is this always reasonable? i.e. Are all SQLi the same?***
  - Can not know without a technical/objective analysis of the vulnerability/threat



Vulnerability Summary for CVE-2016-2174

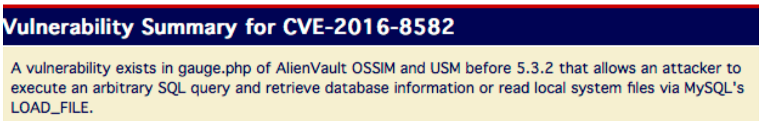
Original release date: 06/13/2016

Last revised: 06/14/2016

Source: US-CERT/NIST

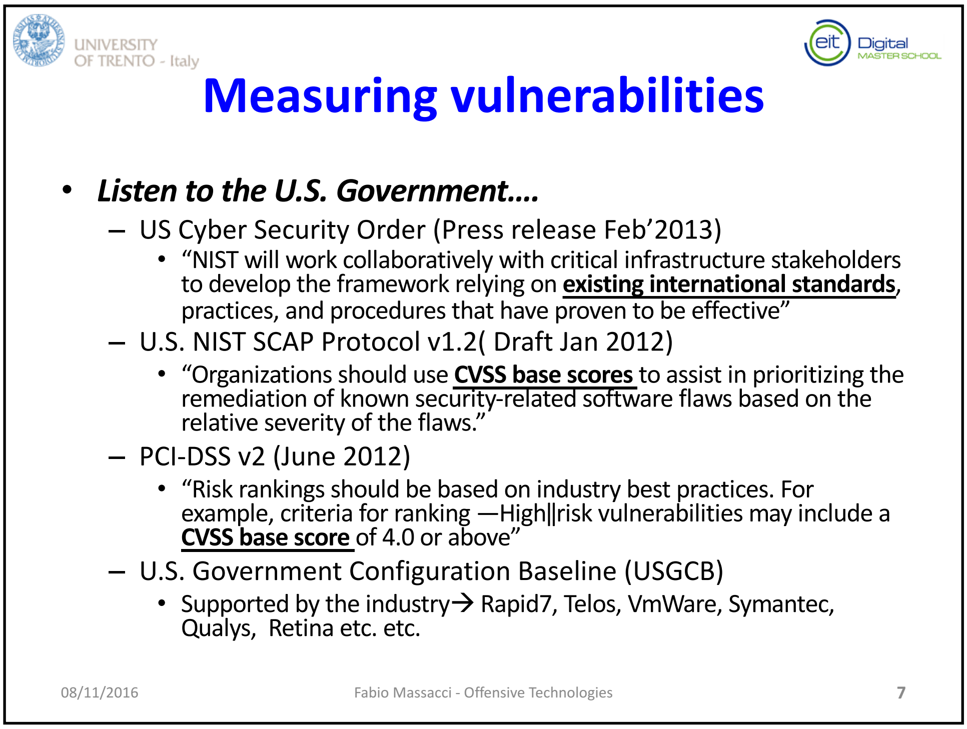
Overview

SQL injection vulnerability in the policy admin tool in Apache Ranger before 0.5.3 allows remote authenticated administrators to execute arbitrary SQL commands via the eventTime parameter to service/plugins/policies/eventTime.



Vulnerability Summary for CVE-2016-8582

A vulnerability exists in gauge.php of AlienVault OSSIM and USM before 5.3.2 that allows an attacker to execute an arbitrary SQL query and retrieve database information or read local system files via MySQL's LOAD_FILE.



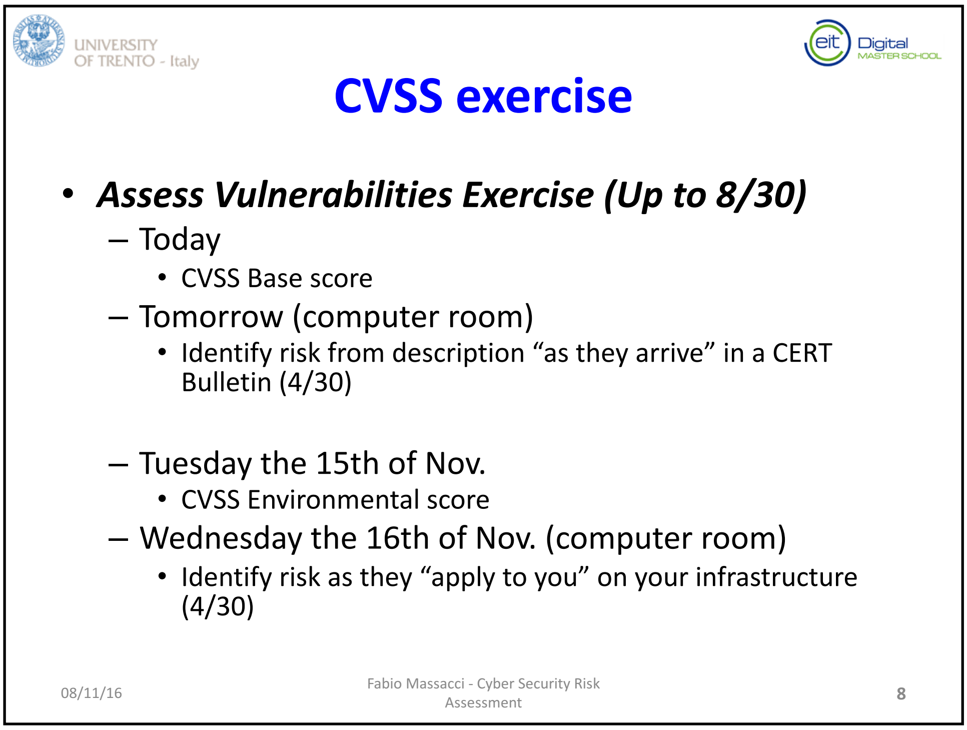
UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Measuring vulnerabilities

- ***Listen to the U.S. Government….***
  - US Cyber Security Order (Press release Feb'2013)
    - "NIST will work collaboratively with critical infrastructure stakeholders to develop the framework relying on **existing international standards**, practices, and procedures that have proven to be effective"
  - U.S. NIST SCAP Protocol v1.2(Draft Jan 2012)
    - "Organizations should use **CVSS base scores** to assist in prioritizing the remediation of known security-related software flaws based on the relative severity of the flaws."
  - PCI-DSS v2 (June 2012)
    - "Risk rankings should be based on industry best practices. For example, criteria for ranking —High||risk vulnerabilities may include a **CVSS base score** of 4.0 or above"
  - U.S. Government Configuration Baseline (USGCB)
    - Supported by the industry→ Rapid7, Telos, VmWare, Symantec, Qualys, Retina etc. etc.

08/11/2016 Fabio Massacci - Offensive Technologies 7



UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

CVSS exercise

- ***Assess Vulnerabilities Exercise (Up to 8/30)***
  - Today
    - CVSS Base score
  - Tomorrow (computer room)
    - Identify risk from description "as they arrive" in a CERT Bulletin (4/30)
  - Tuesday the 15th of Nov.
    - CVSS Environmental score
  - Wednesday the 16th of Nov. (computer room)
    - Identify risk as they "apply to you" on your infrastructure (4/30)

08/11/16 Fabio Massacci - Cyber Security Risk Assessment 8

UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

CVSS – A FRAMEWORK TO QUANTIFY VULNERABILITY SEVERITY

UNIVERSITY OF TRENTO - Italy

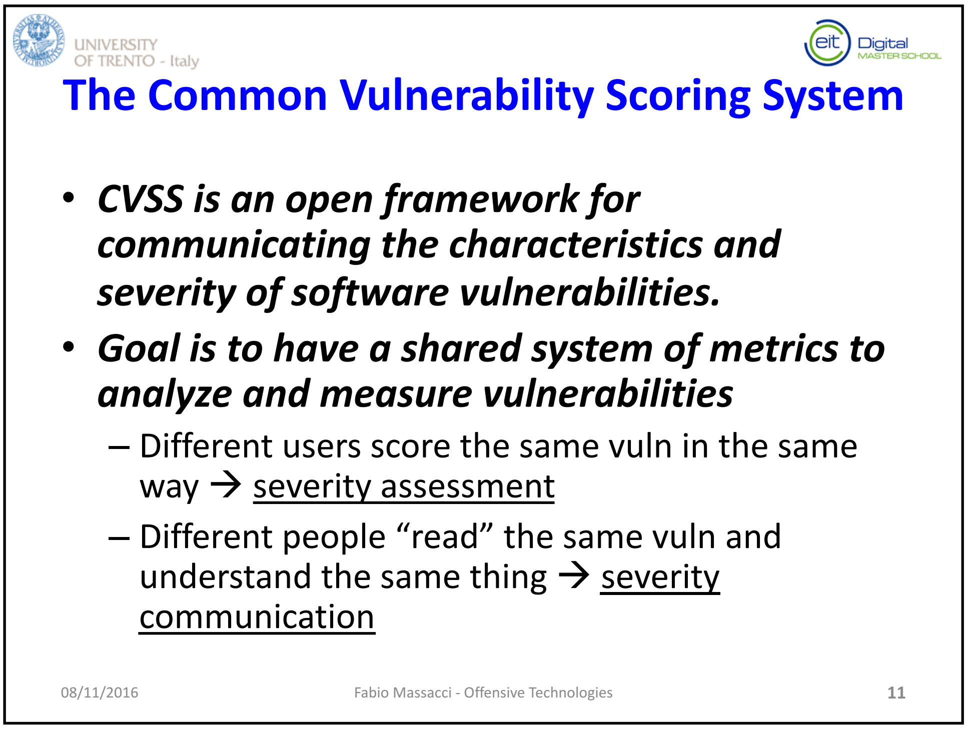
eit Digital MASTER SCHOOL

Vulnerabilities

- ***A flaw or weakness in system security procedures, design, implementation, or internal controls that could be exercised (accidentally triggered or intentionally exploited) and result in a security breach or a violation of the system's security policy***

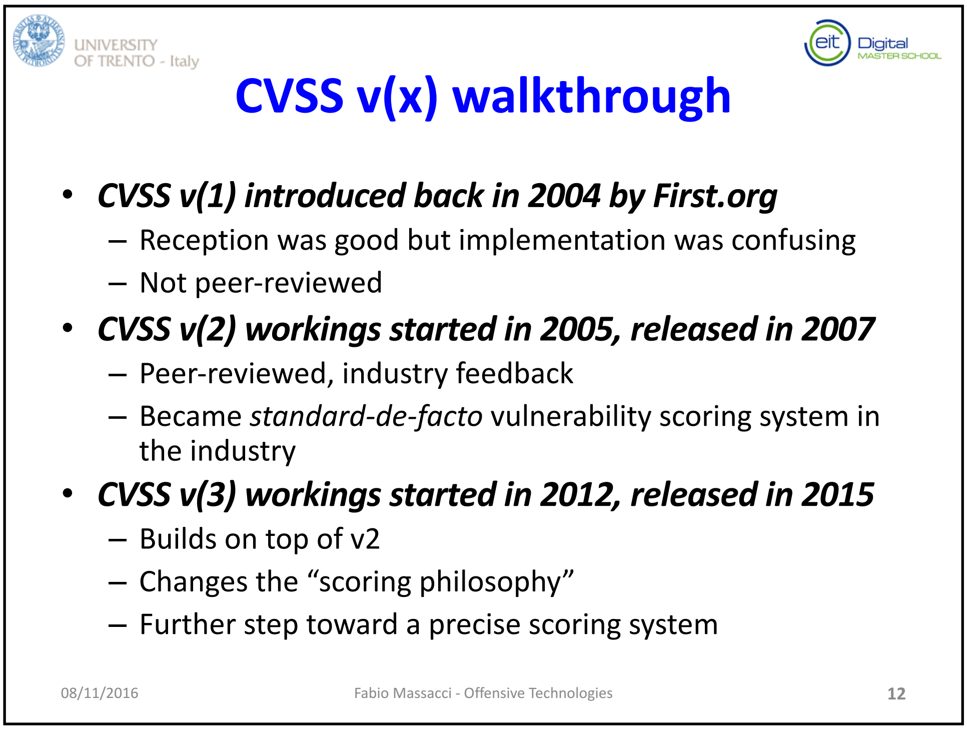
  Definition from NIST SP 800-30

- ***Software vulnerabilities***
  - Buffer overflows
  - Authentication
  - Privilege escalation
  - XSS
  - …



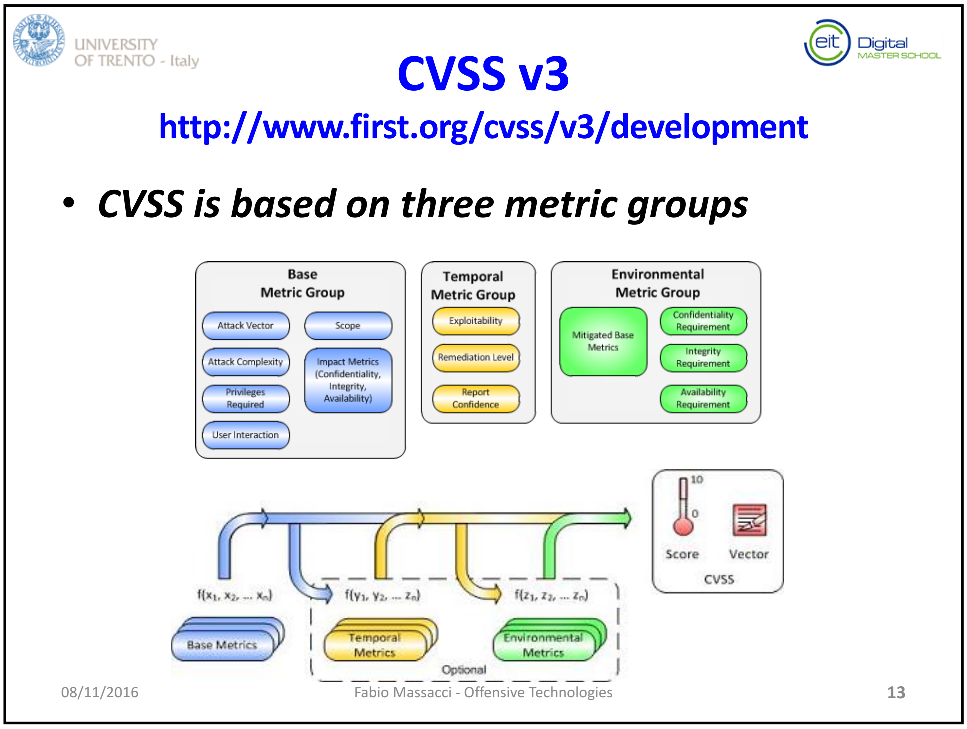
The Common Vulnerability Scoring System

- ***CVSS is an open framework for communicating the characteristics and severity of software vulnerabilities.***
- ***Goal is to have a shared system of metrics to analyze and measure vulnerabilities***
  - Different users score the same vuln in the same way → <u>severity assessment</u>
  - Different people "read" the same vuln and understand the same thing → <u>severity communication</u>



CVSS v(x) walkthrough

- ***CVSS v(1) introduced back in 2004 by First.org***
  - Reception was good but implementation was confusing
  - Not peer-reviewed
- ***CVSS v(2) workings started in 2005, released in 2007***
  - Peer-reviewed, industry feedback
  - Became *standard-de-facto* vulnerability scoring system in the industry
- ***CVSS v(3) workings started in 2012, released in 2015***
  - Builds on top of v2
  - Changes the "scoring philosophy"
  - Further step toward a precise scoring system



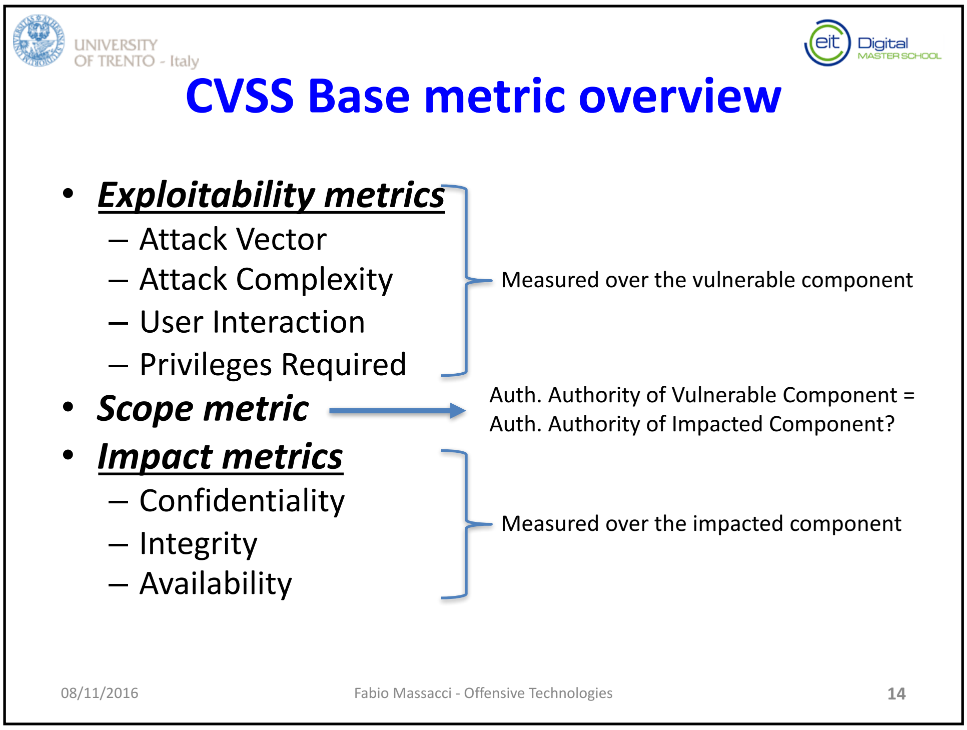
UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

CVSS v3

http://www.first.org/cvss/v3/development

- ***CVSS is based on three metric groups***



UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

CVSS Base metric overview

- ***Exploitability metrics***
  - Attack Vector
  - Attack Complexity
  - User Interaction
  - Privileges Required

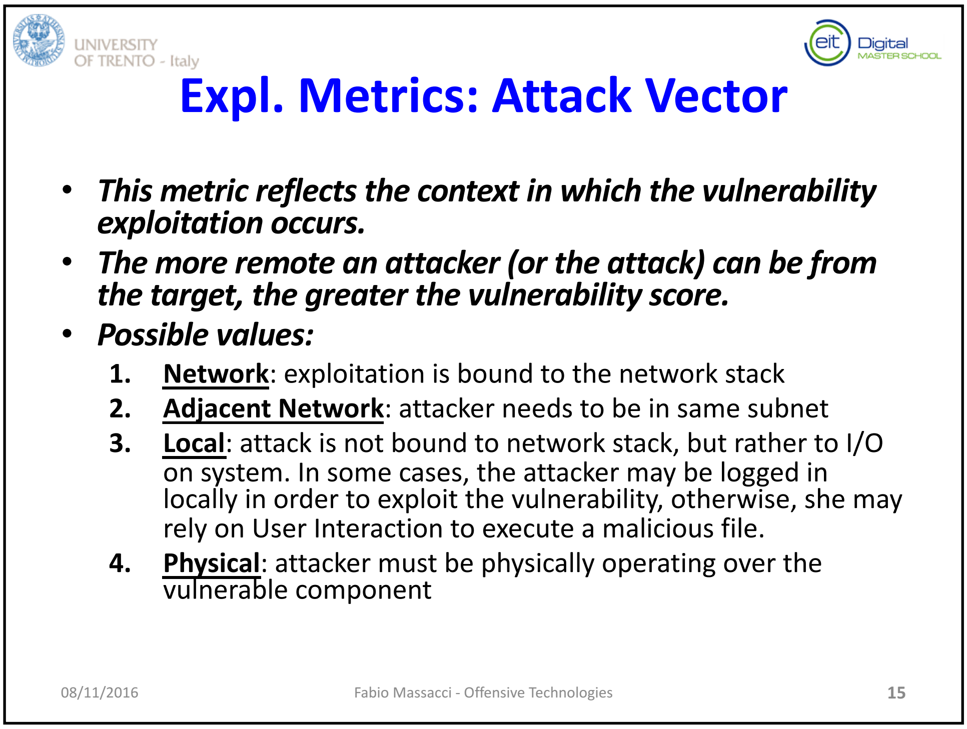
Measured over the vulnerable component

- ***Scope metric*** →

Auth. Authority of Vulnerable Component =
Auth. Authority of Impacted Component?

- ***Impact metrics***
  - Confidentiality
  - Integrity
  - Availability

Measured over the impacted component



UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Expl. Metrics: Attack Vector

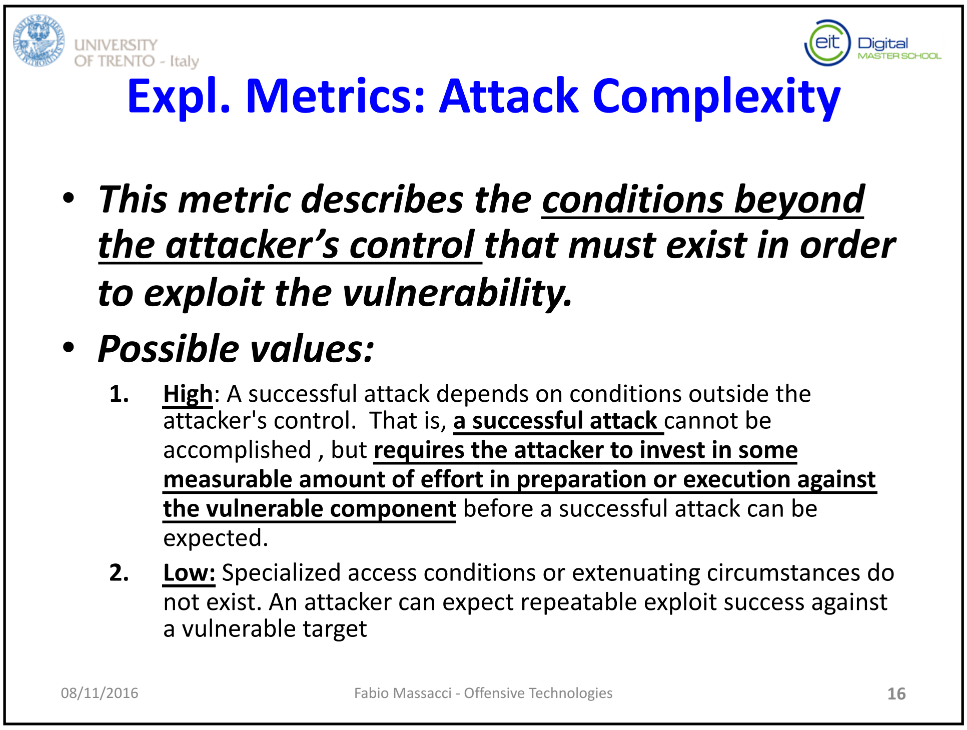
- ***This metric reflects the context in which the vulnerability exploitation occurs.***
- ***The more remote an attacker (or the attack) can be from the target, the greater the vulnerability score.***
- ***Possible values:***

  1. **Network**: exploitation is bound to the network stack

  2. **Adjacent Network**: attacker needs to be in same subnet

  3. **Local**: attack is not bound to network stack, but rather to I/O on system. In some cases, the attacker may be logged in locally in order to exploit the vulnerability, otherwise, she may rely on User Interaction to execute a malicious file.

  4. **Physical**: attacker must be physically operating over the vulnerable component



UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Expl. Metrics: Attack Complexity

- ***This metric describes the conditions beyond the attacker's control that must exist in order to exploit the vulnerability.***
- ***Possible values:***

  1. **High**: A successful attack depends on conditions outside the attacker's control. That is, **a successful attack** cannot be accomplished , but **requires the attacker to invest in some measurable amount of effort in preparation or execution against the vulnerable component** before a successful attack can be expected.

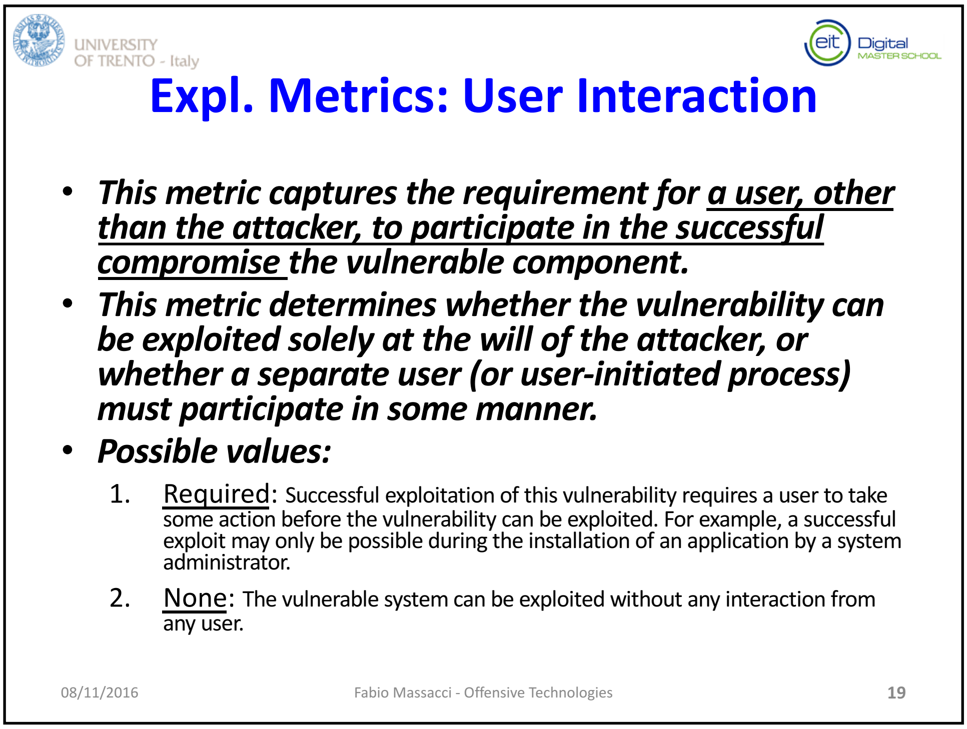
  2. **Low:** Specialized access conditions or extenuating circumstances do not exist. An attacker can expect repeatable exploit success against a vulnerable target

Examples for Attack Complexity: High

- ***For example, a successful attack may depend on an attacker overcoming any of the following conditions:***
  1. The attacker must conduct **target-specific reconnaissance**. For example, on target configuration settings, sequence numbers, shared secrets, etc.
  2. The attacker must **prepare the target environment** to improve exploit reliability. For example, repeated exploitation to win a race condition, or overcoming advanced exploit mitigation techniques.
  3. The attacker **injects herself into the logical network path** between the target and the resource requested by the victim in order to read and/or modify network communications (e.g. man in the middle attack).

Expl. Metrics: Privileges Required

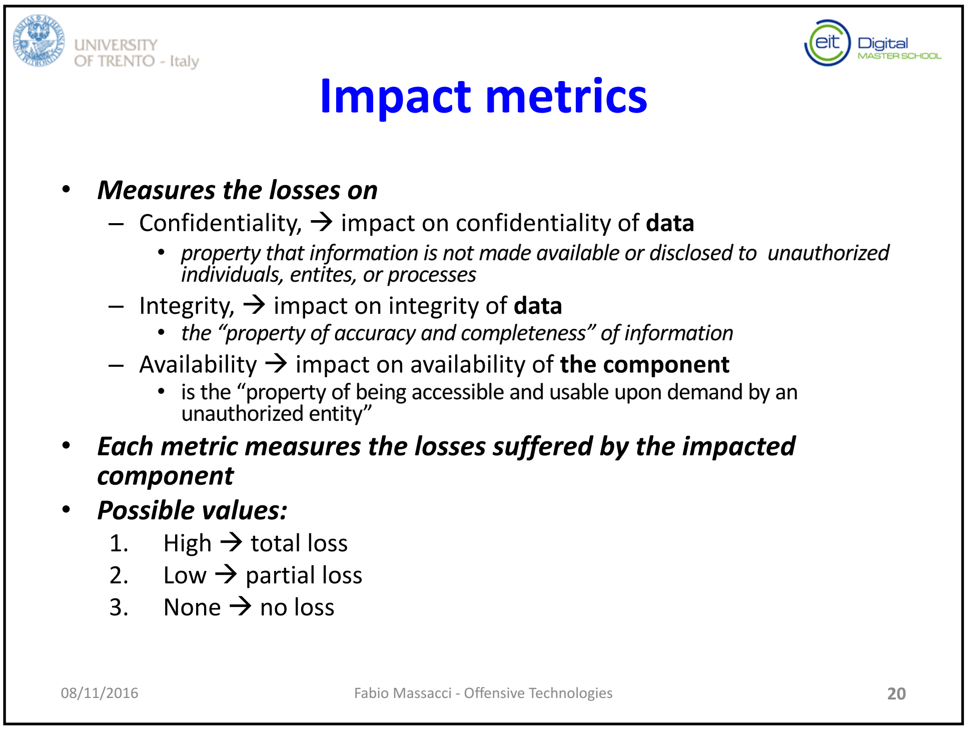
- ***This metric describes the level of privileges an attacker must possess before successfully exploiting the vulnerability.***
- ***Possible values:***
  1. High: The attacker is authorized with (i.e. requires) privileges that provide significant (e.g. administrative) control over the vulnerable component that could affect component-wide settings and files.
  2. Low: The attacker is authorized with (i.e. requires) privileges that provide basic user capabilities that could normally affect only settings and files owned by a user. Alternatively, an attacker with Low privileges may have the ability to cause an impact only to non-sensitive resources.
  3. None: The attacker is unauthorized prior to attack, and therefore does not require any access to settings or files to carry out an attack.



UNIVERSITY OF TRENTO - Italy

Expl. Metrics: User Interaction

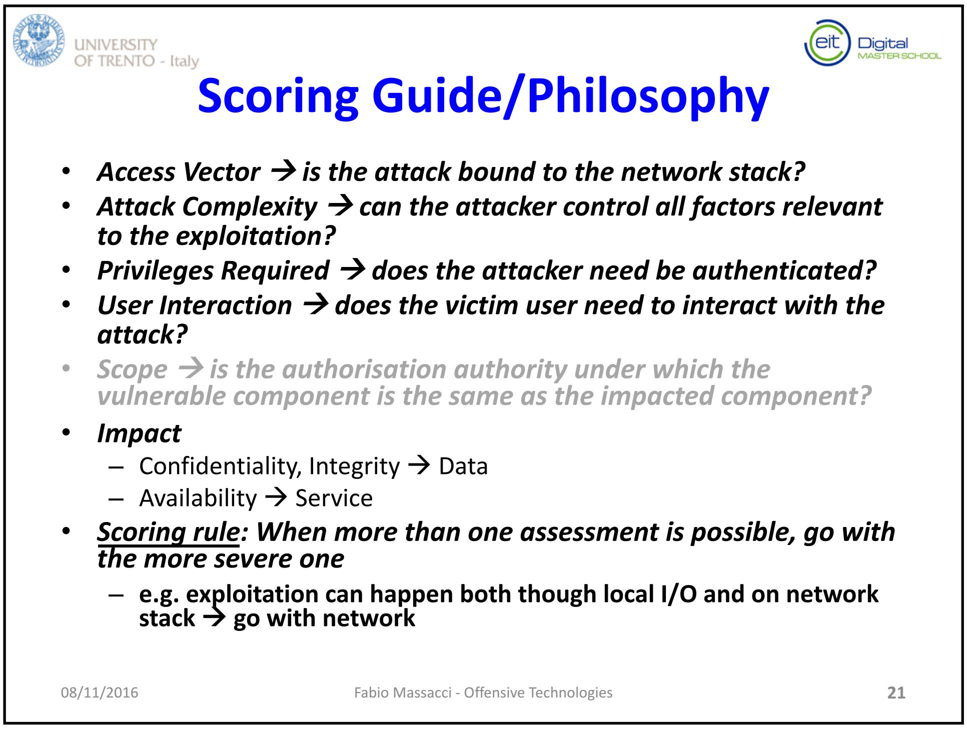
- ***This metric captures the requirement for <u>a user, other than the attacker, to participate in the successful compromise</u> the vulnerable component.***
- ***This metric determines whether the vulnerability can be exploited solely at the will of the attacker, or whether a separate user (or user-initiated process) must participate in some manner.***
- ***Possible values:***
  1. <u>Required</u>: Successful exploitation of this vulnerability requires a user to take some action before the vulnerability can be exploited. For example, a successful exploit may only be possible during the installation of an application by a system administrator.
  2. <u>None</u>: The vulnerable system can be exploited without any interaction from any user.



UNIVERSITY OF TRENTO - Italy

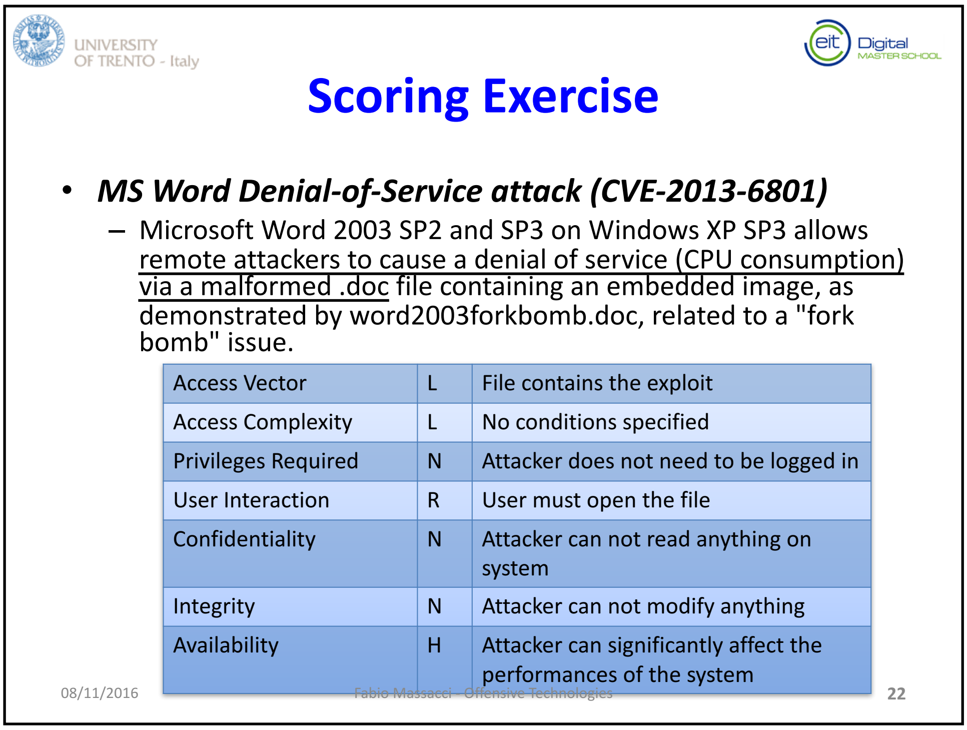
Impact metrics

- ***Measures the losses on***
  - Confidentiality, → impact on confidentiality of **data**
    - *property that information is not made available or disclosed to unauthorized individuals, entites, or processes*
  - Integrity, → impact on integrity of **data**
    - *the "property of accuracy and completeness" of information*
  - Availability → impact on availability of **the component**
    - is the "property of being accessible and usable upon demand by an unauthorized entity"
- ***Each metric measures the losses suffered by the impacted component***
- ***Possible values:***
  1. High → total loss
  2. Low → partial loss
  3. None → no loss



Scoring Guide/Philosophy

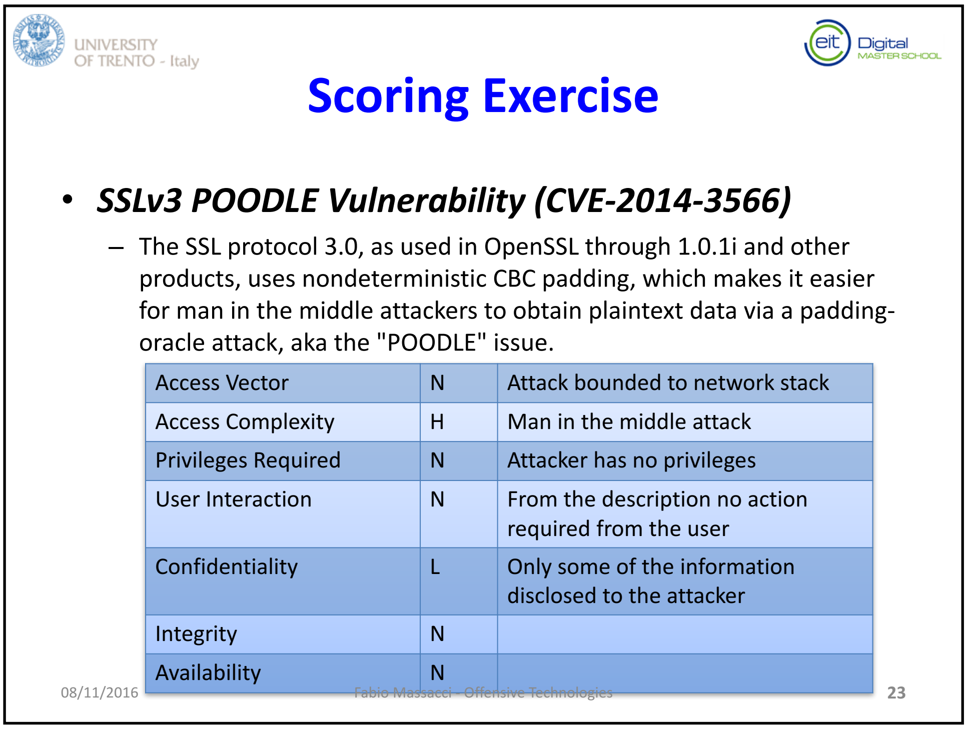
- ***Access Vector* → *is the attack bound to the network stack?***
- ***Attack Complexity* → *can the attacker control all factors relevant to the exploitation?***
- ***Privileges Required* → *does the attacker need be authenticated?***
- ***User Interaction* → *does the victim user need to interact with the attack?***
- ***Scope* → *is the authorisation authority under which the vulnerable component is the same as the impacted component?***
- ***Impact***
  - Confidentiality, Integrity → Data
  - Availability → Service
- ***Scoring rule: When more than one assessment is possible, go with the more severe one***
  - **e.g. exploitation can happen both though local I/O and on network stack → go with network**



Scoring Exercise

- ***MS Word Denial-of-Service attack (CVE-2013-6801)***
  - Microsoft Word 2003 SP2 and SP3 on Windows XP SP3 allows remote attackers to cause a denial of service (CPU consumption) via a malformed .doc file containing an embedded image, as demonstrated by word2003forkbomb.doc, related to a "fork bomb" issue.

Access Vector	L	File contains the exploit
Access Complexity	L	No conditions specified
Privileges Required	N	Attacker does not need to be logged in
User Interaction	R	User must open the file
Confidentiality	N	Attacker can not read anything on system
Integrity	N	Attacker can not modify anything
Availability	H	Attacker can significantly affect the performances of the system



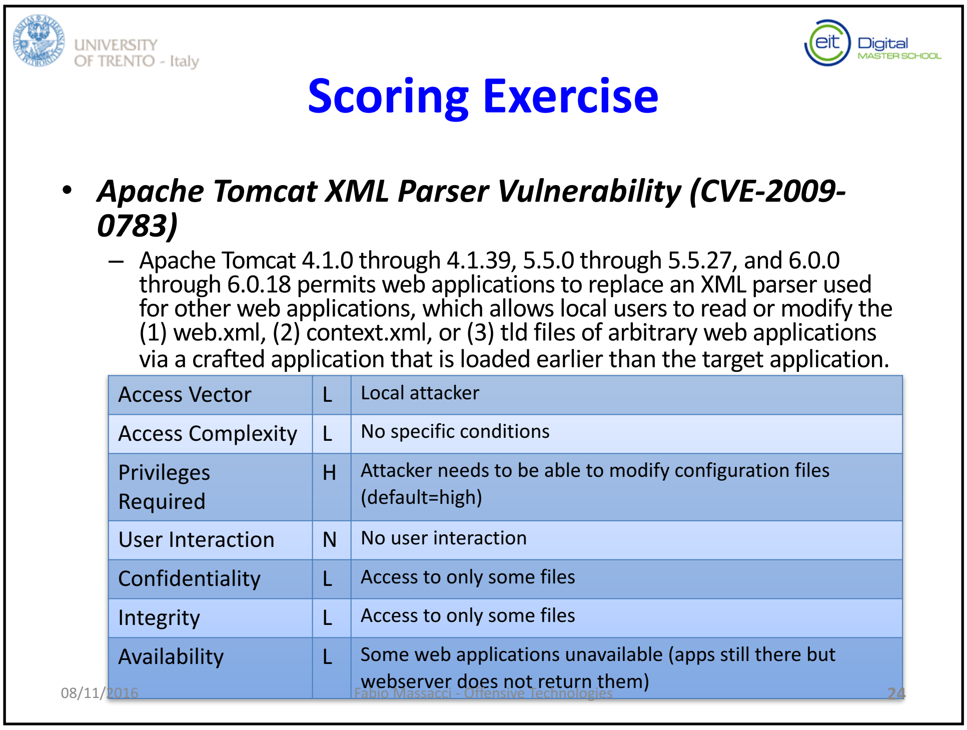
UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Scoring Exercise

- ***SSLv3 POODLE Vulnerability (CVE-2014-3566)***
  – The SSL protocol 3.0, as used in OpenSSL through 1.0.1i and other products, uses nondeterministic CBC padding, which makes it easier for man in the middle attackers to obtain plaintext data via a padding-oracle attack, aka the "POODLE" issue.

Access Vector	N	Attack bounded to network stack
Access Complexity	H	Man in the middle attack
Privileges Required	N	Attacker has no privileges
User Interaction	N	From the description no action required from the user
Confidentiality	L	Only some of the information disclosed to the attacker
Integrity	N	
Availability	N	



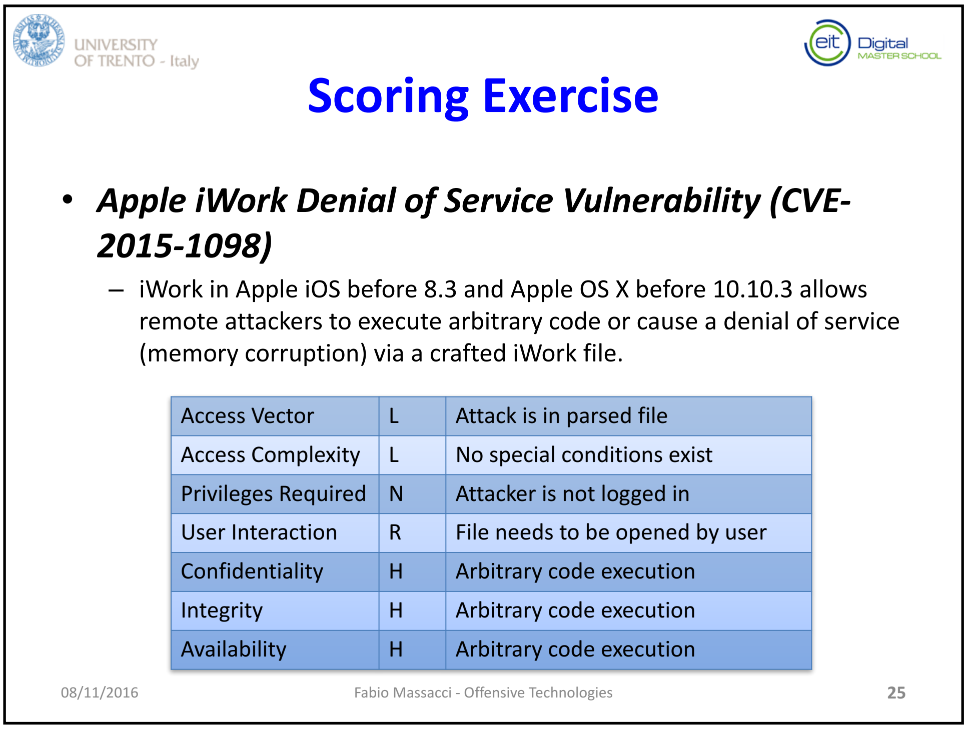
UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Scoring Exercise

- ***Apache Tomcat XML Parser Vulnerability (CVE-2009-0783)***
  – Apache Tomcat 4.1.0 through 4.1.39, 5.5.0 through 5.5.27, and 6.0.0 through 6.0.18 permits web applications to replace an XML parser used for other web applications, which allows local users to read or modify the (1) web.xml, (2) context.xml, or (3) tld files of arbitrary web applications via a crafted application that is loaded earlier than the target application.

Access Vector	L	Local attacker
Access Complexity	L	No specific conditions
Privileges Required	H	Attacker needs to be able to modify configuration files (default=high)
User Interaction	N	No user interaction
Confidentiality	L	Access to only some files
Integrity	L	Access to only some files
Availability	L	Some web applications unavailable (apps still there but webserver does not return them)



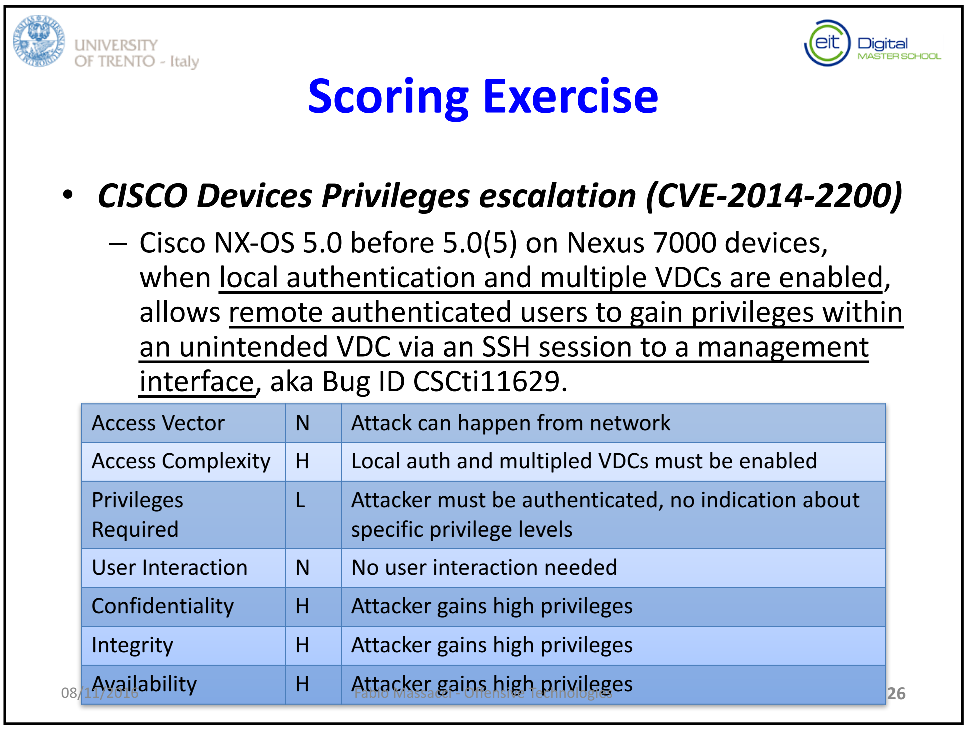
UNIVERSITY OF TRENTO - Italy

Digital MASTER SCHOOL

Scoring Exercise

- ***Apple iWork Denial of Service Vulnerability (CVE-2015-1098)***
  - iWork in Apple iOS before 8.3 and Apple OS X before 10.10.3 allows remote attackers to execute arbitrary code or cause a denial of service (memory corruption) via a crafted iWork file.

Access Vector	L	Attack is in parsed file
Access Complexity	L	No special conditions exist
Privileges Required	N	Attacker is not logged in
User Interaction	R	File needs to be opened by user
Confidentiality	H	Arbitrary code execution
Integrity	H	Arbitrary code execution
Availability	H	Arbitrary code execution



UNIVERSITY OF TRENTO - Italy

Digital MASTER SCHOOL

Scoring Exercise

- ***CISCO Devices Privileges escalation (CVE-2014-2200)***
  - Cisco NX-OS 5.0 before 5.0(5) on Nexus 7000 devices, when local authentication and multiple VDCs are enabled, allows remote authenticated users to gain privileges within an unintended VDC via an SSH session to a management interface, aka Bug ID CSCti11629.

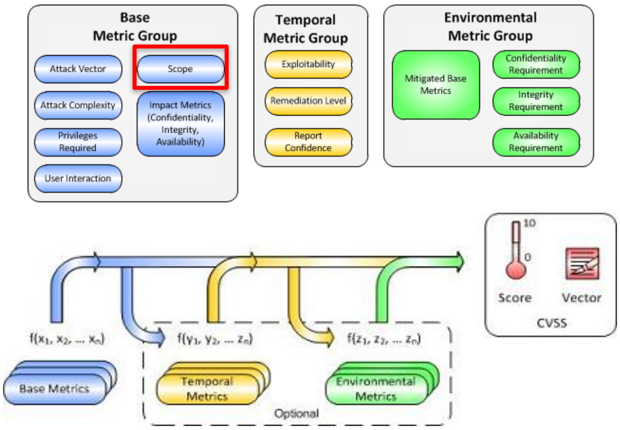
Access Vector	N	Attack can happen from network
Access Complexity	H	Local auth and multipled VDCs must be enabled
Privileges Required	L	Attacker must be authenticated, no indication about specific privilege levels
User Interaction	N	No user interaction needed
Confidentiality	H	Attacker gains high privileges
Integrity	H	Attacker gains high privileges
Availability	H	Attacker gains high privileges

SCOPE METRIC

CVSS v3

http://www.first.org/cvss/v3/development

- ***CVSS is based on three metric groups***



UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

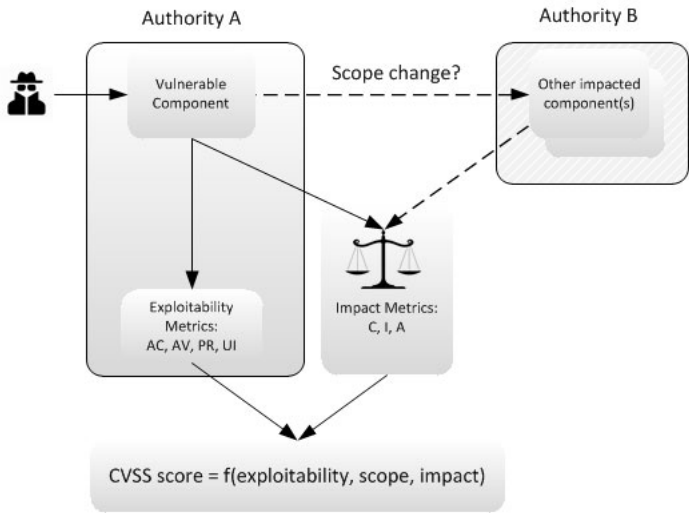
Scope (1)

- *Scope refers to the <u>collection of privileges defined by a computing authority</u> (e.g. an application, an operating system, or a sandbox environment) when granting access to computing resources (e.g. files, CPU, memory, etc). These privileges are assigned based on some method of identification and authorization.*
- *When the <u>vulnerability</u> of a software component governed by one authorization scope <u>is able to affect resources governed by another authorization scope, a Scope change has occurred.</u>*

UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Scope (2)


Authority A
Authority B
Vulnerable Component
Scope change?
Other impacted component(s)
Exploitability Metrics: AC, AV, PR, UI
Impact Metrics: C, I, A
CVSS score = f(exploitability, scope, impact)

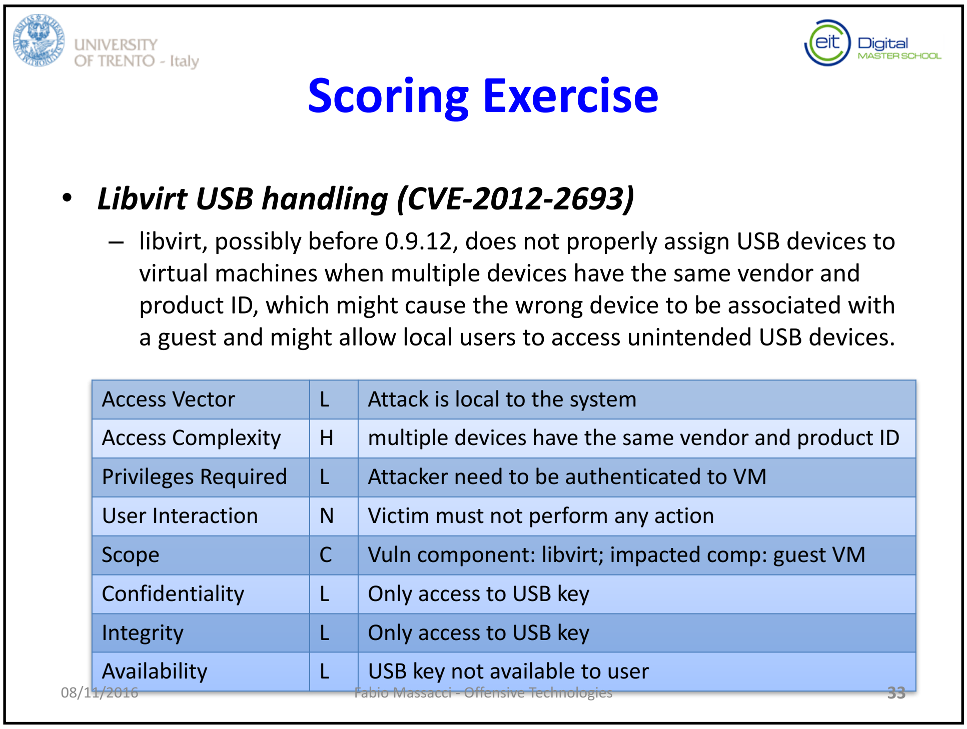
Scope (3)

- ***Possible values:***
  - Unchanged: An exploited vulnerability can only affect resources managed by the same authority. In this case the vulnerable component and the impacted component are the same.
  - Changed: An exploited vulnerability can affect resources beyond the authorization privileges intended by the vulnerable component. In this case the vulnerable component and the impacted component are different.

Scoring Exercise

- ***CISCO host crash (CVE-2011-0355)***
  - Cisco Nexus 1000V Virtual Ethernet Module (VEM) 4.0(4) SV1(1) through SV1(3b), as used in VMware ESX 4.0 and 4.1 and ESXi 4.0 and 4.1, does not properly handle dropped packets, which allows guest OS users to cause a denial of service (ESX or ESXi host OS crash) by sending an 802.1Q tagged packet over an access vEthernet port, aka Cisco Bug ID CSCtj17451.

Access Vector	N/A	Virtual ports typically from adjacent network
Access Complexity	L	No specific conditions
Privileges Required	N	Forging a network packet does not required privileges on vuln system
User Interaction	N	No user interaction
Scope	C	Vulnerable component=guest OS; impacted component=host OS;
Confidentiality	N	Host crash only
Integrity	N	Host crash only
Availability	H	Host crash



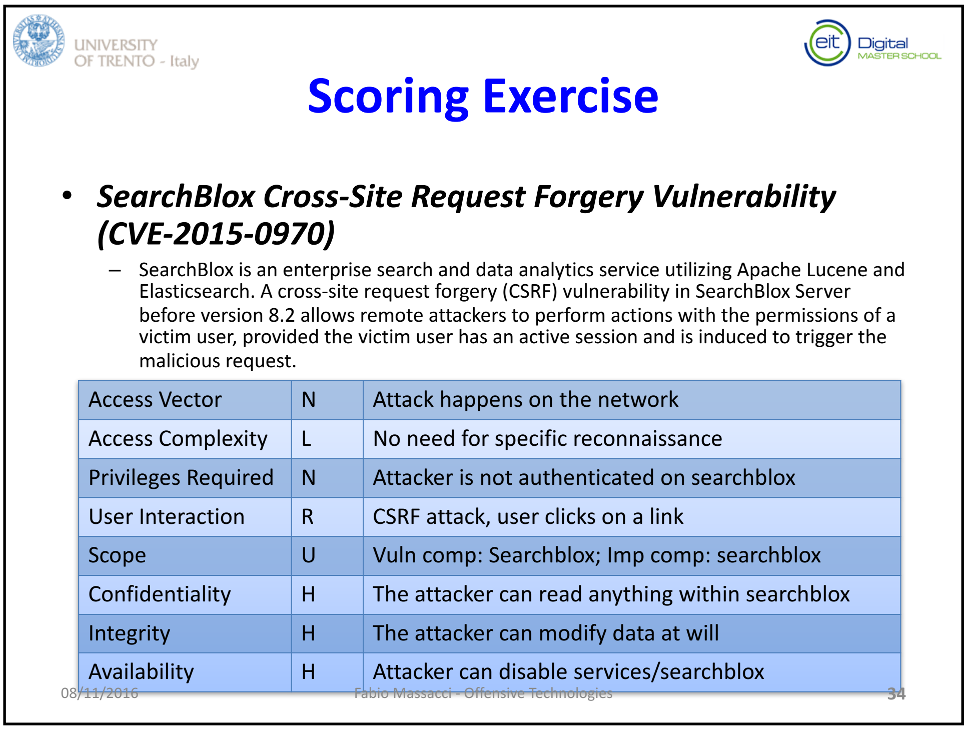
UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Scoring Exercise

- ***Libvirt USB handling (CVE-2012-2693)***
  - libvirt, possibly before 0.9.12, does not properly assign USB devices to virtual machines when multiple devices have the same vendor and product ID, which might cause the wrong device to be associated with a guest and might allow local users to access unintended USB devices.

Access Vector	L	Attack is local to the system
Access Complexity	H	multiple devices have the same vendor and product ID
Privileges Required	L	Attacker need to be authenticated to VM
User Interaction	N	Victim must not perform any action
Scope	C	Vuln component: libvirt; impacted comp: guest VM
Confidentiality	L	Only access to USB key
Integrity	L	Only access to USB key
Availability	L	USB key not available to user



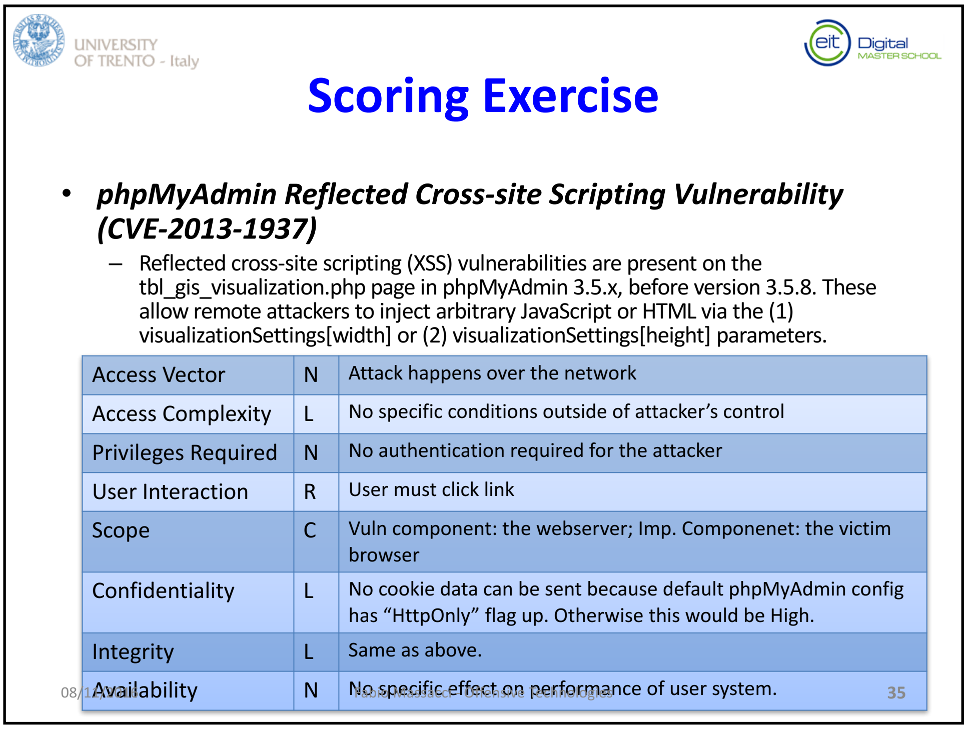
UNIVERSITY OF TRENTO - Italy

eit Digital MASTER SCHOOL

Scoring Exercise

- ***SearchBlox Cross-Site Request Forgery Vulnerability (CVE-2015-0970)***
  - SearchBlox is an enterprise search and data analytics service utilizing Apache Lucene and Elasticsearch. A cross-site request forgery (CSRF) vulnerability in SearchBlox Server before version 8.2 allows remote attackers to perform actions with the permissions of a victim user, provided the victim user has an active session and is induced to trigger the malicious request.

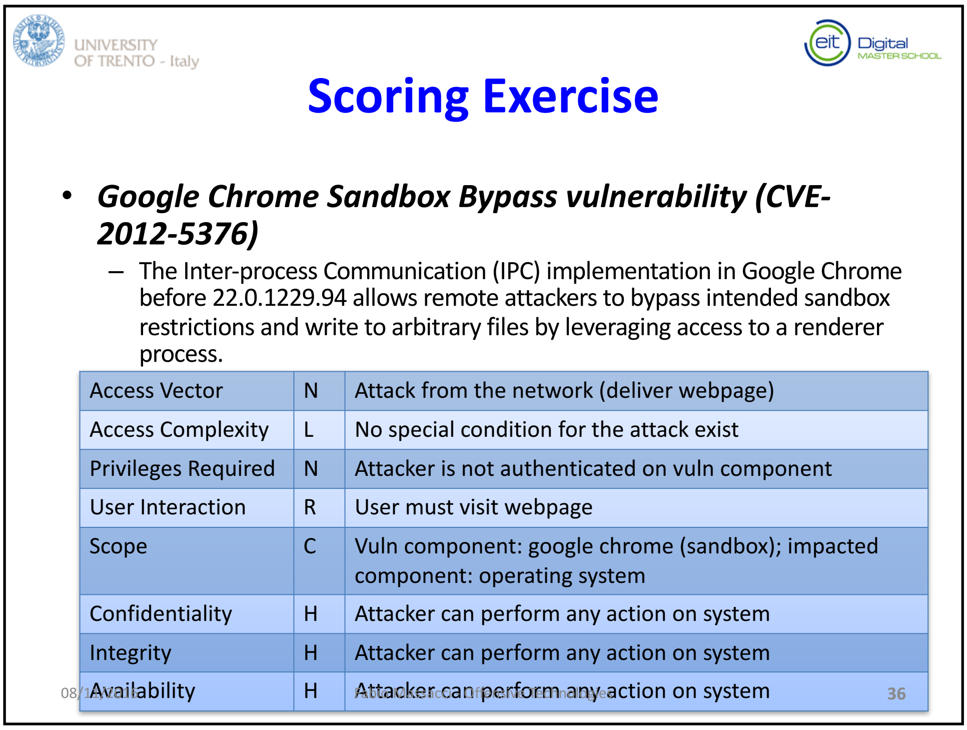
Access Vector	N	Attack happens on the network
Access Complexity	L	No need for specific reconnaissance
Privileges Required	N	Attacker is not authenticated on searchblox
User Interaction	R	CSRF attack, user clicks on a link
Scope	U	Vuln comp: Searchblox; Imp comp: searchblox
Confidentiality	H	The attacker can read anything within searchblox
Integrity	H	The attacker can modify data at will
Availability	H	Attacker can disable services/searchblox



Scoring Exercise

- ***phpMyAdmin Reflected Cross-site Scripting Vulnerability (CVE-2013-1937)***
  - Reflected cross-site scripting (XSS) vulnerabilities are present on the tbl_gis_visualization.php page in phpMyAdmin 3.5.x, before version 3.5.8. These allow remote attackers to inject arbitrary JavaScript or HTML via the (1) visualizationSettings[width] or (2) visualizationSettings[height] parameters.

Access Vector	N	Attack happens over the network
Access Complexity	L	No specific conditions outside of attacker's control
Privileges Required	N	No authentication required for the attacker
User Interaction	R	User must click link
Scope	C	Vuln component: the webserver; Imp. Componenet: the victim browser
Confidentiality	L	No cookie data can be sent because default phpMyAdmin config has “HttpOnly” flag up. Otherwise this would be High.
Integrity	L	Same as above.
Availability	N	No specific effect on performance of user system.



Scoring Exercise

- ***Google Chrome Sandbox Bypass vulnerability (CVE-2012-5376)***
  - The Inter-process Communication (IPC) implementation in Google Chrome before 22.0.1229.94 allows remote attackers to bypass intended sandbox restrictions and write to arbitrary files by leveraging access to a renderer process.

Access Vector	N	Attack from the network (deliver webpage)
Access Complexity	L	No special condition for the attack exist
Privileges Required	N	Attacker is not authenticated on vuln component
User Interaction	R	User must visit webpage
Scope	C	Vuln component: google chrome (sandbox); impacted component: operating system
Confidentiality	H	Attacker can perform any action on system
Integrity	H	Attacker can perform any action on system
Availability	H	Attacker can perform any action on system