Vulnerabilities

JOÃO PAULO BARRACA

1

Vulnerabilities (definition)

Is a weakness in a system (software, hardware…)
- It's a broad concept as a vulnerability can have multiple origins and causes

Vulnerabilities allow an attacker to violate a reasonable security policy for a system
- Policies define how a system should behave.
- Examples:
  - Wheels will turn left only when steering wheel turns left
  - Phones will only allow access to its owner
  - Programs will only run code inserted by its original developer

Vulnerability number always increases as software grows
- It's inherent to the increased complexity, interactions, development process
  - Also, they do not disappear
- Software is updated with fixes, but older software is still vulnerable and may be present

2

Vulnerabilities (definition, cont)

Vulnerabilities are states in a computing system that either allows an attacker to:

1. execute commands as another user
2. access data that is contrary to the specified access restrictions for that data
3. pose as another entity
4. conduct a denial of service (DoS) (affect availability)

3

A simple vulnerability - secura.com


Blogs

Last month, Microsoft patched a very interesting vulnerability that would allow an attacker with a foothold on your internal network to essentially become Domain Admin with one click. All that is required is for a connection to the Domain Controller to be possible from the attacker's viewpoint.

Secura's security expert Tom Tervoort previously discovered a less severe Netlogon vulnerability last year that allowed workstations to be taken over, but the attacker required a Person-in-the-Middle (PitM) position for that to work. Now, he discovered this second, much more severe (CVSS score: 10.0) vulnerability in the protocol. By forging an authentication token for specific Netlogon functionality, he was able to call a function to set the computer password of the Domain Controller to a known value. After that, the attacker can use this new password to take control over the domain controller and steal credentials of a domain admin.

The vulnerability stems from a flaw in a cryptographic authentication scheme used by the Netlogon Remote Protocol, which among other things can be used to update computer passwords. This flaw allows attackers to impersonate any computer, including the domain controller itself, and execute remote procedure calls on their behalf.

4

CIA triad – What vulnerabilities directly impact

Confidentiality
- Whether information is disclosed to others

Integrity
- Whether data contents and formats are kept safe from modifications

Availability
- Whether system performance is degraded

5

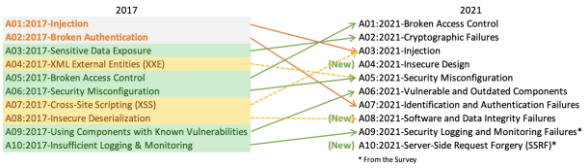
Vulnerability sources – OWASP Top 10 (Web)

1. Broken Access Control
2. Cryptographic Failures
3. Injection
4. Insecure Design
5. Security Misconfiguration
6. Vulnerable and Outdated Components
7. Identification and Authentication failures
8. Software and Data Integrity Failures
9. Security Logging and monitoring Failures
10. Server Side Request Forgery

List of the Most Prevalent Sources of Vulnerabilities, as determined by the community. Reevaluated every 3-5 years.

6

Vulnerability sources – OWASP Top 10 (Web)



7

Vulnerability sources – 7 Pernicious Kingdoms

1. **Input Validation and Representation**
2. **API Abuse**
3. **Security Features**
4. **Time and State**
5. **Errors**
6. **Code Quality**
7. **Encapsulation**

***. Environment**

K. Tsipenyuk, B. Chess and G. McGraw, "Seven pernicious kingdoms: a taxonomy of software security errors," in IEEE Security & Privacy, vol. 3, no. 6, pp. 81-84, Nov.-Dec. 2005, doi: 10.1109/MSP.2005.159.

8

Vulnerability sources - CWE

Vulnerabilities may exist due to Bugs or Faults
- Bug is an error in the implementation of a software
- Fault is a design or architectural error

CWE - Common Weakness Enumeration
- Extensive (944) list of anti-patterns that may lead to insecure systems
- Organized in a tree, with examples in multiple languages

9

CWE-348: Use of Less Trusted Source

The software has two different sources of the same data or information, but it uses the source that has less support for verification, is less trusted, or is less resistant to attack.

Details at: https://cwe.mitre.org/data/definitions/348.html
- Describes pattern, provides examples, provides list of related CVEs

10

CWE-348: Use of Less Trusted Source

```
$requestingIP = '0.0.0.0';
if (array_key_exists('HTTP_X_FORWARDED_FOR', $_SERVER)) {
        $requestingIP = $_SERVER['HTTP_X_FORWARDED_FOR'];
} else {
        $requestingIP = $_SERVER['REMOTE_ADDR'];
}

if(in_array($requestingIP,$ipAllowlist)){
        generatePage();
        return;
} else {
        echo "You are not authorized to view this page";
        return;
}
```

Set by Web Server or Client

Set by Web Server

X-Forwarded-For is a comma-separated list: the client can prepend entries, so trust it only from known proxies.

11

Vulnerability Tracking by vendors

During the development cycle, vulnerabilities are handled as bugs
- May have been handled by a security team or not
- May have a security classification and SLA

When software is available, vulnerabilities are also tracked globally
- For every system and software publicly available

Public tracking helps…
- focusing the discussion around the same issue
  - Ex: a library that is used in multiple applications, distributions
- defenders to easily test their systems, enhancing the security
- attackers to easily know what vulnerability can be used

12

Vulnerability Tracking

Vulnerabilities are privately tracked
- Constitute an arsenal for future attacks against targets
- Exploits can be considered as assets in cyberwar

Knowledge about vulnerabilities and exploits is publicly traded
- Prices range from near zero to several million USD, through direct markets and acquisition programmes
- Two distinct markets: public bug bounty programmes (e.g. Google, Microsoft, Pwn2Own) and private acquisition/brokers (e.g. Zerodium, Crowdfense)
  - Public bug bounty programmes: typically hundreds to tens of thousands USD; top contest awards in the high six figures
  - Private brokers publish price lists in USD, up to several million for a one-click mobile chain; lists change often
  - Many others (E.g. https://www.crowdfense.com/exploit-acquisition-program/)

…and privately traded at unknown prices
- Private Companies, Organized Crime, APTs

Private exploit market: published acquisition prices

Source, https://www.crowdfense.com/exploit-acquisition-program/
(prices change often - check the current list)

Vulnerability Tracking

Most well-known trackers systems: CVE and NVD
- CVE: Common Vulnerabilities and Exposures, managed by MITRE
- NVD: National Vulnerability Database, managed by NIST
  - Fed by CVE@MITRE but provides enhanced information
  - Note: NVD enrichment (CVSS, CWE) is currently delayed/incomplete - rely on the CNA-provided CVSS and the vendor advisory

Others
- CERT Vulnerability Notes Database (VNDB)
  - Maintained by CERTs, may provide additional information regarding a CVE
- VulnDB
  - Focus on APIs and providing information to companies
- DISA IAVA and STIGs
  - Information Assurance Vulnerability Alerts: includes MIL and GOV systems
  - Security Technical Implementation Guides
- Industry Sharing and Analysis Centers (ISAC)
  - Industry driven, thematic (AUTO, FINANCIAL, IT, etc. groups)

CVE: Common Vulnerabilities and Exposures

Dictionary of publicly known information security vulnerabilities and exposures
- For vulnerability management
- For patch management
- For vulnerability alerting
- For intrusion detection

Uses common identifiers for the same CVEs
- Enable data exchange between security products
- Provide a baseline index point for evaluating coverage of tools and services.

Details about a vulnerability can be kept private
- Part of responsible disclosure
- Until 2022: Until vendor provides a fix
- After 2022: Published after 90 days, even if the vendor has no patch to fix the product

CNA: CVE Numbering Authorities Policy

CNAs SHOULD assign CVE IDs to Vulnerabilities whether or not a Fix is available.
- Hard publication deadlines tied to disclosure, not patch status

CNAs SHOULD publish within 24 hours of Publicly Disclosing an assigned CVE ID.

CNAs MUST publish within 72 hours of Publicly Disclosing.

The Secretariat MAY publicly identify the CNA who reserved the CVE ID 24 hours after a CVE ID has been Publicly Disclosed.

CNAs MUST ensure that a public reference exists on the internet before or concurrently with publication
- A CVE Record MUST NOT be the first Public Disclosure of a Vulnerability.

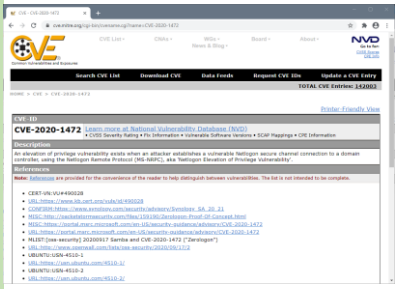
For Publicly Disclosed vulnerabilities, if the scoped CNA doesn't respond/refuses within 72 hours, a Root MUST direct another CNA/CNA-LR to assign "as quickly as possible and no later than 72 hours."

CVE-2020-1472

@MITRE

Basic information about the CVE

References to other trackers (provided for convenience)



CVE-2020-1472

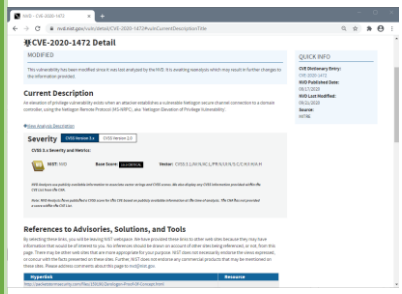
@NVD

Basic information about the CVE and a small analysis of it

The CVE Severity Score

Links to advisories, solutions

Note: NVD has a substantial backlog and is not up to date - CVEs may lack CVSS/CWE analysis.



19

CVE-2020-1472

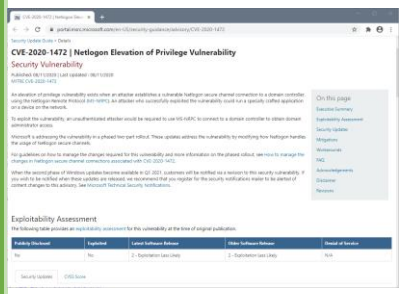
@Product Owner

More detail, why it happens, and how it can be mitigated

Information about patches/updates available to help IT staff and users

Information about its exploitability.

Format is vendor dependent. Each vendor defines what/how to show information



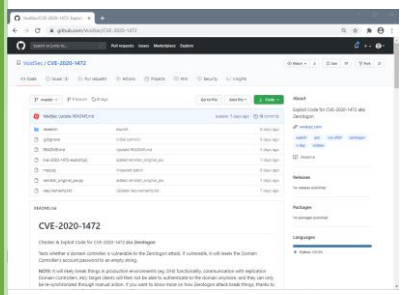
20

CVE-2020-1472

@Other places

Independent researchers may publish validation tools or exploits

Very dynamic community with public and private facets



21

Vulnerability tracking

Not an easy task

- Exploits are not always known
- Impact and Value may be underestimated

Old feeds may create a false sense of security

A highly dynamic community is great…

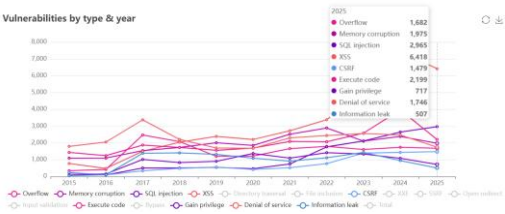
- To defenders as they can test and implement defenses
- To attackers as they can incorporate exploits

22

CVE per year (1/2) - cvedetails.com, as of Sep 2025

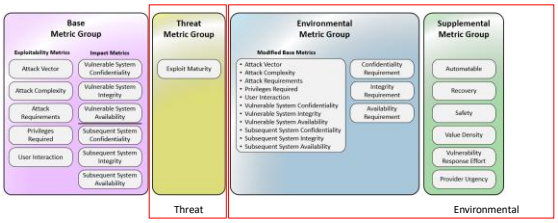
23

CVE per year (2/2) - cvedetails.com, as of Sep 2025



24

CVSS - Common Vulnerability Scoring System (v4.0)



25

CVSS – Common Vulnerability Scoring System

Base metrics
- Metric intrinsic to the vulnerability
- How exploitable it is
- What is the potential impact

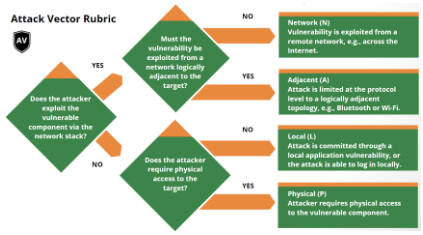
Threat metrics
- What is the current situation regarding the support for its exploitability
- Existence of PoC, active exploits, active campaigns

Environmental metrics
- What is the actual situation at each customer
  - How it really impacts the system operation, and what are the system requirements
- How the vulnerability is relevant to the customer and its clients

Metric groups (v4.0): Base + Threat + Environmental. In v3.1 and v2: Base + Temporal + Environmental.

26

CVSS – Common Vulnerability Scoring System



27

CVSS - Common Vulnerability Scoring System (v4.0): worked example

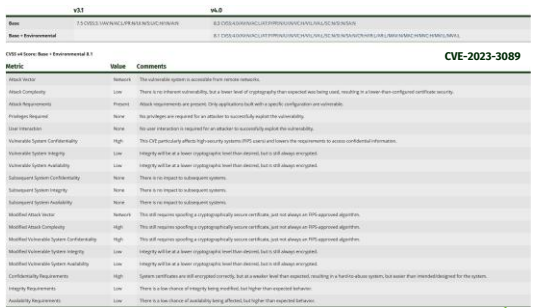
Provides a quick way to determine the severity of a vulnerability (0-10 score)
- Helps defenders prioritizing the deployment of mitigations
- Helps attackers selecting the most convenient vulnerability to explore

Example: 7.3 - CVSS:4.0/AV:L/AC:L/AT:P/PR:L/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N

Attack Vector	Local	An attacker must be able to access the vulnerable system with a local, interactive session.
Attack Complexity	Low	No specialized conditions or advanced knowledge are required.
Attack Requirements	Present	Multiple conditions that require target-specific reconnaissance and preparation must be satisfied in order to achieve successful exploitation of this vulnerability.
Privileges Required	Low	An attacker must be able to place a file within the web root to be processed by NGINX.
User Interaction	None	No user interaction is required for an attacker to successfully exploit the vulnerability.
Vulnerable System Confidentiality	High	The attacker could execute arbitrary code on the vulnerable system with elevated privileges.
Vulnerable System Integrity	High	The attacker could execute arbitrary code on the vulnerable system with elevated privileges.
Vulnerable System Availability	High	The attacker could execute arbitrary code on the vulnerable system with elevated privileges.
Subsequent System Confidentiality	None	There is no impact to the subsequent system confidentiality.
Subsequent System Integrity	None	There is no impact to the subsequent system integrity.
Subsequent System Availability	None	There is no impact to the subsequent system availability.

https://www.first.org/cvss/calculator/4.0

28

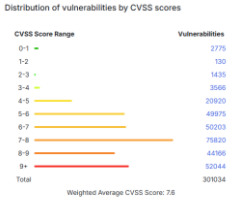


29

CVSS - Common Vulnerability Scoring System (v4.0): from score to remediation SLA

Risk Ranking	CVSS Score (v3.1 / v4.0)	SLA in days
Critical	9.0 – 10.0	15
High	7.0 – 8.9	30
Medium	4.0 – 6.9	90
Low	0.1 – 3.9	180
Informational (FIRST: None)	0.0	None

Vulnerability Management SLAs must consider the CVSS score, the customer's capability to
Score bands = FIRST official severity ratings; the SLA days are an illustrative internal policy.



30

Vulnerability Disclosure

How should a researcher act when a vulnerability is found?

If the engagement is private: deliver to the contracting entity
- May request the public release of the information…
- Commonly handled under a Non-Disclosure Agreement (NDA)

What about other cases?

31

Vulnerability Disclosure: None

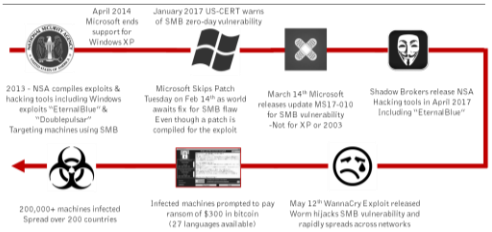
Researcher doesn't notify vendor about vulnerability
- Doesn't care
- Uses it as part of an arsenal or trades the information

Leads to 0-day vulnerabilities
- Vulnerability is not known to the public and there is no direct remediation
- Some other third parties may also know about the vulnerability and exploit it

If impact is high, it creates major disruption when publicly known
- Quick adoption in malware and dissemination
  - Remember: Windows systems may take at least one month to be patched

32

CVE-2017-0144 EternalBlue



Source undetermined

33

Vulnerability Disclosure: Coordinated

1. Researcher informs vendor about vulnerability and impact
- Usually through a form of report with estimation of impact and/or demonstration

2. Vendor implements and distributes a correction
- But not always!

3. Vulnerability is mostly fixed in supported systems

Optional: CVE entry is requested: https://cveform.mitre.org/

Optional: A website with a sound name is created for public awareness
- Heartbleed, Shellshock, CRIME, POODLE, Spectre, Log4Shell, Dirty Cow…

34

CVE-2020-15802 – Sep 9 2020

https://hexhive.epfl.ch/BLURtooth/

Researcher:
- "We discovered the vulnerability in March 2020 and responsibly disclosed our findings along with suggested countermeasures to the Bluetooth SIG in May 2020. We kept our findings private and the Bluetooth SIG publicly disclosed them, without informing us, on the 10th of September of 2020. Our work is assigned CVE-2020-15802."

Bluetooth SIG:
- At the time of writing, there are no deployed patches to address the BLUR attacks on actual devices. The Bluetooth SIG suggested that version 5.1 of the standard will contain guidelines to mitigate the BLUR attacks (e.g., disable key overwrites in certain circumstances as proposed in our countermeasures), but such guidelines are not (yet) public and we cannot comment on them. The Bluetooth SIG provides a public statement about BLURtooth and the BLUR attacks.

35

Vulnerability Disclosure: Full

Researcher discloses the vulnerability without warning
- As a CVE
- In a public mailing list
- As a blog entry, webpage or news item
- As an exploit

Vendor is pressured to issue a fix as soon as possible
- But not always
  - It doesn't!
  - It considers the product not supported
  - It under-reports the issue

Some mayhem may occur until a fix is applied
- Remember all those phones/TVs/etc… without frequent updates

36

How to disclose

CSIRT Teams define a formal document for the organization
- Follows RFC 2350
- https://www.ua.pt/pt/ciberseguranca/rfc2350
- Contains information about relevant networks, responsible person and contact points
- Signed with PGP

/security.txt defines a formal document for the domain
- Follows RFC 9116
- Machine-parsable format describing their vulnerability disclosure practices to make it easier for researchers to report vulnerabilities

37

How to disclose

Through the relevant CERT
- https://www.cncs.gov.pt/pt/certpt/
- CERT will inform the target organization directly

https://www.openbugbounty.org/
- Non-profit platform connecting security researchers and domain owners
- Doesn't imply a payout, but it allows a negotiation

Through contact addresses made available by the entities

38

Key takeaways (1/2)

A vulnerability is a state **that lets someone violate the security policy** you expect of a system

Weakness vs incident: CWE names the weakness families, CVE names the specific vulnerability/exposure

Trackers are not all equal:
- CVE = identifiers
- NVD/CNAs = enriched data
  - NVD enrichment is currently delayed - do not assume a CVE has a score, or take conclusions because it hasn't a score
- vendor portals = remediation.

The same CVE is seen differently by MITRE, NVD, the vendor and the wild: follow all four views

39

Key takeaways (2/2)

CVSS v4.0 scores severity, not risk
- Use it as one input, alongside threat/KEV signals and your own Environmental context

Scoring only gets you a ranking
- Remediation SLAs must also weigh the customer's capability and the product context

Disclosure models differ in who controls the timeline.
- Coordinated disclosure is the professional default;
- Full disclosure is a political statement

Before reporting:
- find the target's policy (/security.txt, CSIRT/CERT document, CNA) and put your contact details in writing

40

References & further reading

- FIRST, "Common Vulnerability Scoring System v4.0" specification & reference guide, first.org/cvss/v4.0
- FIRST, "CVSS v3.1 Specification Guide", 2019 - useful for comparing metric groups across versions
- MITRE, "CWE - Common Weakness Enumeration", cwe.mitre.org
- CVE Program, cve.org - CNA & Root policies (publication deadlines, record states); CVE Request form: cveform.mitre.org
- NIST, National Vulnerability Database, nvd.nist.gov - watch the published notes on the enrichment backlog
- OWASP, "Top 10:2025 - The Ten Most Critical Web Application Security Risks", owasp.org/
- K. Tsipenyuk, B. Chess, G. McGraw, "Seven Pernicious Kingdoms: A Taxonomy of Software Security Errors", IEEE Security & Privacy 3(6), Nov-Dec 2005, doi:10.1109/MSP.2005.159
- CISA, Known Exploited Vulnerabilities (KEV) catalogue, cisa.gov/known-exploited-vulnerabilities-catalog
- FIRST EPSS (first.org/epss) and SSVC (first.org/ssvc) - alternatives to score-only prioritisation
- RFC 2350 "A CSIRT Services Framework"
- RFC 9116 "security.txt: A Mechanism for Computer Security Disclosure and Contact Information"
- HexHive (EPFL), "BLURtooth", hexhive.epfl.ch/BLURtooth - disclosure case study

41