

#1 - Vulnerabilities

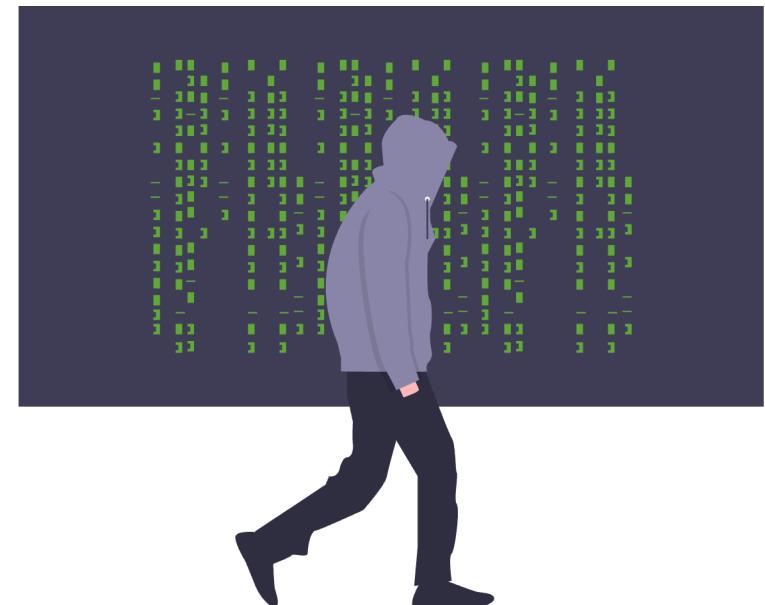
Vulnerabilities

Is a weakness in a system (software, hardware...)

- It's a broad concept as a vulnerability can derive from many things

A vulnerability allows an attacker to violate a reasonable security policy for that 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
 - User roles
 - access levels



Vulnerabilities

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



Vulnerabilities

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

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



Threats



CIA triad

→ Model for security systems

Confidentiality

- Whether information is disclosed to others
 - ↳ 2 factor authentication

Integrity

- Whether data contents and formats are kept safe from modifications
 - ↳ Encryption

Availability

- Whether system performance is degraded
 - ↳ safe against Dos

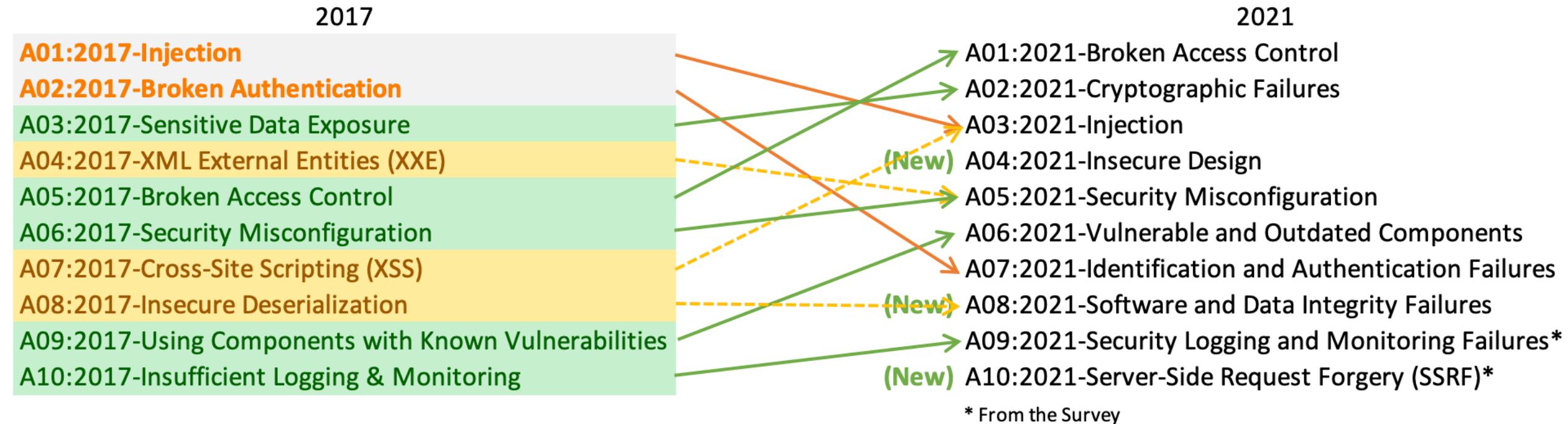


Vulnerability sources – OWASP Top 10 (Web)

- 1. Injection**
- 2. Broken Authentication**
- 3. Sensitive Data Exposure**
- 4. XML External Entities (XXE)**
- 5. Broken Access control**
- 6. Security misconfigurations**
- 7. Cross Site Scripting (XSS)**
- 8. Insecure Deserialization**
- 9. Using Components with known vulns.**
- 10. Insufficient logging and monitoring**



Vulnerability sources – OWASP Top 10 (Web)



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. Tsiptenyuk, 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.



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 Weaknesses Enumeration

- Extensive (891) list of anti-patterns that may lead to insecure systems
- Arranged in a tree, with examples in multiple languages

→ padrões met. é específico
a app mcs sim ao seu pro
produz e corr.
e produzido

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

É mais abstrato que o CVE. Contém os padrões de uma vulnerabilidade.



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

Vulnerability Tracking by vendors

During the development cycle, vulnerabilities are handled as bugs

- May have a dedicated security team or not

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



Vulnerability Tracking

Vulnerabilities are privately tracked

- Constitute an arsenal for future attacks against targets
- Exploits are weapons

Knowledge about vulnerabilities and exploits is publicly traded

- From 0 to 2-3M€ (more?) through direct markets, or acquisition programs
- Up to 2.5M€ for bug hunting programs or direct acquisition (Google, Zerodium)
 - 2.5M€: 1 click Android exploit
 - 2M€: 1 click iPhone exploit
 - 1.5M€: WhatsApp or iMessage exploit
 - ~2K for a XSS at HackerOne (although there are records of \$1M payouts)

...and privately traded at unknown prices

- Private Companies, Organized Crime, APTs



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

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, FINANTIAL, 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

vulnerability tracking by version, OS, tool, ... less abstract than CWE

Uses common identifiers for the same CVE's

- 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 owner provides a fix



CVE-2020-1472

@MITRE

Basic information about the CVE

References to other trackers (provided for convenience)

The screenshot shows a web browser displaying the MITRE Common Vulnerabilities and Exposures (CVE) database. The URL in the address bar is cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-1472. The page title is "CVE - CVE-2020-1472". The header includes links for "CVE List", "CNAs", "WGs News & Blog", "Board", "About", and the "NVD Go to for: CVSS Scores CPE Info" section. Below the header is a navigation bar with links for "Search CVE List", "Download CVE", "Data Feeds", "Request CVE IDs", and "Update a CVE Entry". A total count of "TOTAL CVE Entries: 142003" is displayed. The main content area shows the details for CVE-2020-1472, including its ID, a brief description, and a list of references from various sources.

CVE-ID

CVE-2020-1472 [Learn more at National Vulnerability Database \(NVD\)](#)
• CVSS Severity Rating • Fix Information • Vulnerable Software Versions • SCAP Mappings • CPE Information

Description

An elevation of privilege vulnerability exists when an attacker establishes a vulnerable Netlogon secure channel connection to a domain controller, using the Netlogon Remote Protocol (MS-NRPC), aka 'Netlogon Elevation of Privilege Vulnerability'.

References

Note: [References](#) are provided for the convenience of the reader to help distinguish between vulnerabilities. The list is not intended to be complete.

- CERT-VN:VU#490028
- [URL:https://www.kb.cert.org/vuls/id/490028](https://www.kb.cert.org/vuls/id/490028)
- CONFIRM:https://www.synology.com/security/advisory/Synology_SA_20_21
- MISC:<http://packetstormsecurity.com/files/159190/Zerologon-Proof-Of-Concept.html>
- MISC:<https://portal.msrc.microsoft.com/en-US/security-guidance/advisory/CVE-2020-1472>
- URL:<https://portal.msrc.microsoft.com/en-US/security-guidance/advisory/CVE-2020-1472>
- MLIST:[oss-security] 20200917 Samba and CVE-2020-1472 ("Zerologon")
- URL:<http://www.openwall.com/lists/oss-security/2020/09/17/2>
- UBUNTU:USN-4510-1
- URL:<https://usn.ubuntu.com/4510-1/>
- UBUNTU:USN-4510-2
- URL:<https://usn.ubuntu.com/4510-2/>

CVE-2020-1472

@NVD

Basic information
about the CVE and a
small analysis of it

The CVE Severity Score

Links to advisories,
solutions

NVD - CVE-2020-1472

nvd.nist.gov/vuln/detail/CVE-2020-1472#vulnCurrentDescriptionTitle

CVE-2020-1472 Detail

MODIFIED

This vulnerability has been modified since it was last analyzed by the NVD. It is awaiting reanalysis which may result in further changes to the information provided.

Current Description

An elevation of privilege vulnerability exists when an attacker establishes a vulnerable Netlogon secure channel connection to a domain controller, using the Netlogon Remote Protocol (MS-NRPC), aka 'Netlogon Elevation of Privilege Vulnerability'.

+View Analysis Description

Severity CVSS Version 3.x CVSS Version 2.0

CVSS 3.x Severity and Metrics:

 **NIST: NVD** **Base Score:** 10.0 CRITICAL **Vector:** CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H

NVD Analysts use publicly available information to associate vector strings and CVSS scores. We also display any CVSS information provided within the CVE List from the CNA.

Note: NVD Analysts have published a CVSS score for this CVE based on publicly available information at the time of analysis. The CNA has not provided a score within the CVE List.

References to Advisories, Solutions, and Tools

By selecting these links, you will be leaving NIST webspace. We have provided these links to other web sites because they may have information that would be of interest to you. No inferences should be drawn on account of other sites being referenced, or not, from this page. There may be other web sites that are more appropriate for your purpose. NIST does not necessarily endorse the views expressed, or concur with the facts presented on these sites. Further, NIST does not endorse any commercial products that may be mentioned on these sites. Please address comments about this page to nvd@nist.gov.

Hyperlink	Resource
http://packetstormsecurity.com/files/159190/Zerologon-Proof-Of-Concept.html	

QUICK INFO

CVE Dictionary Entry: CVE-2020-1472
NVD Published Date: 08/17/2020
NVD Last Modified: 09/21/2020
Source: MITRE

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 it's exploitability.

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

The screenshot shows a Microsoft Edge browser window displaying the security advisory for CVE-2020-1472. The URL is portal.msrc.microsoft.com/en-US/security-guidance/advisory/CVE-2020-1472. The page title is "CVE-2020-1472 | Netlogon Elevation of Privilege Vulnerability". The main content area describes the vulnerability as an elevation of privilege issue related to Netlogon secure channel connections. It mentions a phased two-part rollout by Microsoft. A sidebar titled "On this page" lists various sections: Executive Summary, Exploitability Assessment, Security Updates, Mitigations, Workarounds, FAQ, Acknowledgements, Disclaimer, and Revisions. At the bottom, there is a table for "Exploitability Assessment" with columns for Publicly Disclosed, Exploited, Latest Software Release, Older Software Release, and Denial of Service. The table shows "No" for all categories except Denial of Service which is "N/A". Below the table are tabs for "Security Updates" and "CVSS Score".

CVE-2020-1472 | Netlogon Elevation of Privilege Vulnerability

Security Vulnerability

Published: 08/11/2020 | Last Updated : 08/11/2020
MITRE CVE-2020-1472

An elevation of privilege vulnerability exists when an attacker establishes a vulnerable Netlogon secure channel connection to a domain controller, using the Netlogon Remote Protocol (MS-NRPC). An attacker who successfully exploited the vulnerability could run a specially crafted application on a device on the network.

To exploit the vulnerability, an unauthenticated attacker would be required to use MS-NRPC to connect to a domain controller to obtain domain administrator access.

Microsoft is addressing the vulnerability in a phased two-part rollout. These updates address the vulnerability by modifying how Netlogon handles the usage of Netlogon secure channels.

For guidelines on how to manage the changes required for this vulnerability and more information on the phased rollout, see [How to manage the changes in Netlogon secure channel connections associated with CVE-2020-1472](#).

When the second phase of Windows updates become available in Q1 2021, customers will be notified via a revision to this security vulnerability. If you wish to be notified when these updates are released, we recommend that you register for the security notifications mailer to be alerted of content changes to this advisory. See [Microsoft Technical Security Notifications](#).

Exploitability Assessment

The following table provides an [exploitability assessment](#) for this vulnerability at the time of original publication.

Publicly Disclosed	Exploited	Latest Software Release	Older Software Release	Denial of Service
No	No	2 - Exploitation Less Likely	2 - Exploitation Less Likely	N/A

Security Updates CVSS Score

On this page

- Executive Summary
- Exploitability Assessment
- Security Updates
- Mitigations
- Workarounds
- FAQ
- Acknowledgements
- Disclaimer
- Revisions

CVE-2020-1472

@Other places

Independent researchers
may publish validation tools
or exploits

Very dynamic community
with public and private
facets

The screenshot shows a GitHub repository page for `VoidSec/CVE-2020-1472: Exploit`. The repository has 4 stars, 97 forks, and 21 issues. The code tab is selected, showing a list of commits:

File	Commit Message	Time Ago
VoidSec Update README.md	1ba0d90 5 days ago	19 commits
research	exploit	8 days ago
.gitignore	Initial commit	8 days ago
README.md	Update README.md	5 days ago
cve-2020-1472-exploit.py	added reinstall_original_pw	7 days ago
nrpc.py	impacket patch	8 days ago
reinstall_original_pw.py	added reinstall_original_pw	7 days ago
requirements.txt	Update requirements.txt	7 days ago

The README.md file contains the following content:

CVE-2020-1472

Checker & Exploit Code for CVE-2020-1472 aka ZeroLogon

Tests whether a domain controller is vulnerable to the ZeroLogon attack, if vulnerable, it will reset the Domain Controller's account password to an empty string.

NOTE: It will likely break things in production environments (eg. DNS functionality, communication with replication Domain Controllers, etc); target clients will then not be able to authenticate to the domain anymore, and they can only be re-synchronized through manual action. If you want to know more on how ZeroLogon attack break things, thanks to

About
Exploit Code for CVE-2020-1472 aka ZeroLogon

voidsec.com
exploit poc cve-2020 zerologon
n-day voidsec

Readme

Releases
No releases published

Packages
No packages published

Languages
Python 100.0%

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

[View Analysis Description](#)

Severity	CVSS Version 3.x	CVSS Version 2.0
CVSS 3.x Severity and Metrics:		
NVD	NIST: NVD	Base Score: 10.0 CRITICAL
		Vector: CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:C/H:L/H/A:H

Exploitability Assessment

The following table provides an exploitability assessment for this vulnerability at the time of original publication.

Publicly Disclosed	Exploited	Latest Software Release	Older Software Release	Denial of Service
No	No	2 - Exploitation Less Likely	2 - Exploitation Less Likely	N/A

CVE-2020-1472

Checker & Exploit Code for CVE-2020-1472 aka Zerologon

Tests whether a domain controller is vulnerable to the Zerologon attack, if vulnerable, it will resets the Domain Controller's account password to an empty string.

NOTE: It will likely break things in production environments (eg. DNS functionality, communication with replication Domain Controllers, etc); target clients will then not be able to authenticate to the domain anymore, and they can only be re-synchronized through manual action. If you want to know more on how Zerologon attack break things, thanks to

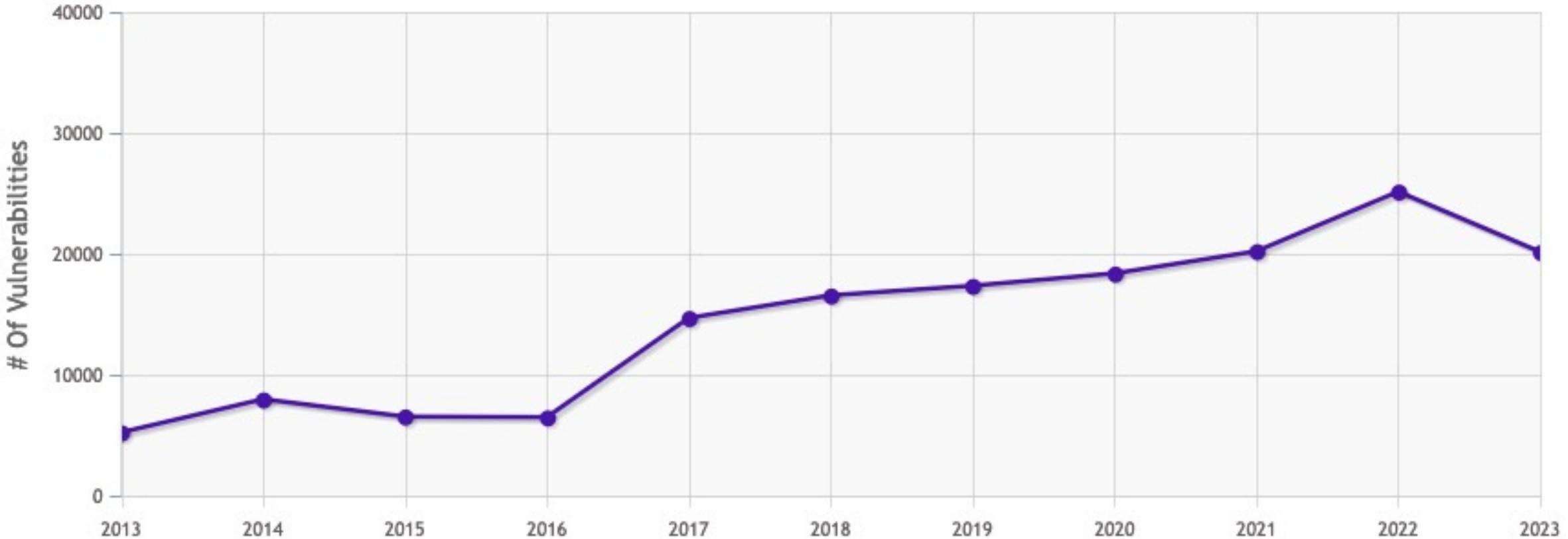
No packages published

Languages

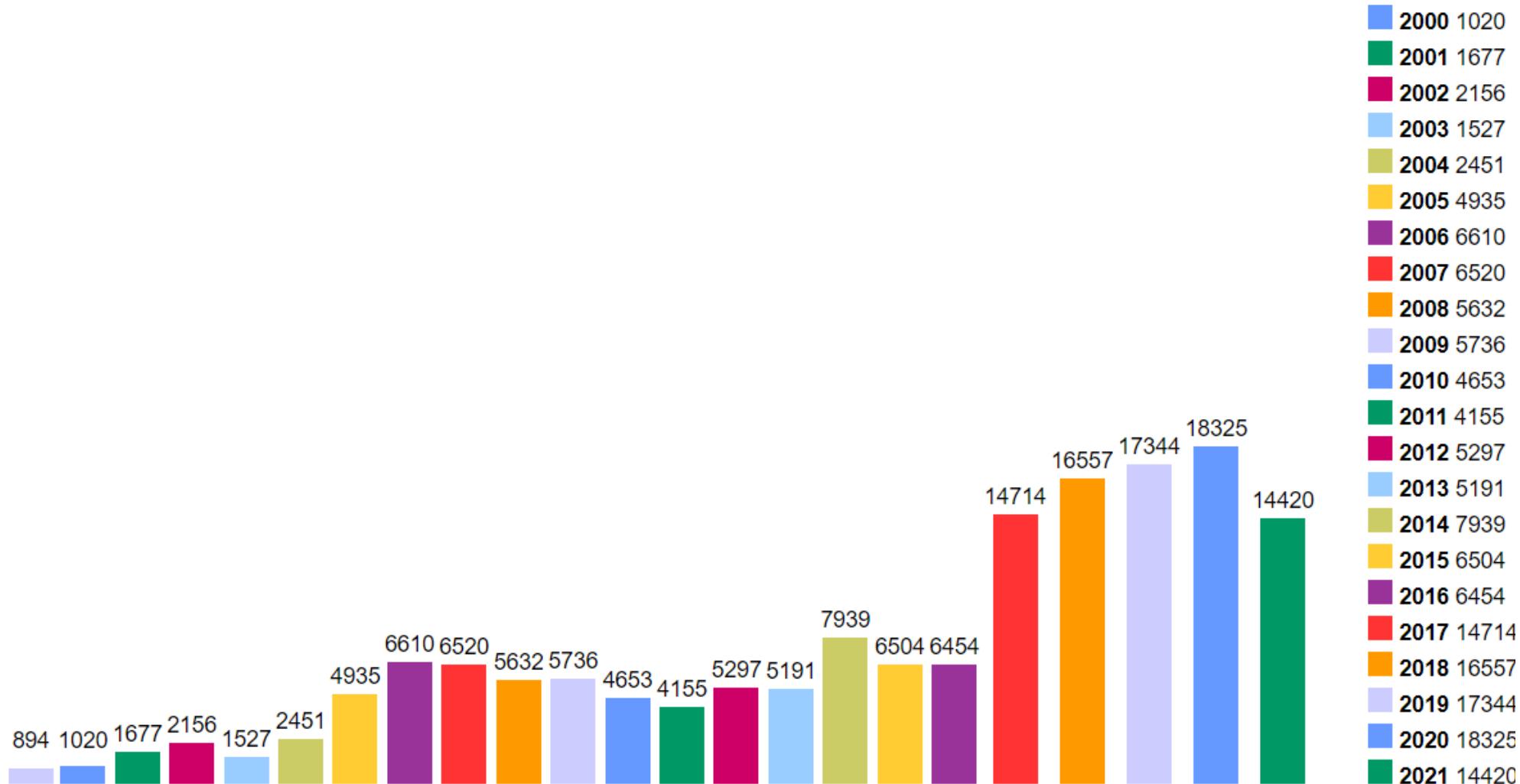
Python 100.0%



CVE per year – cvedetails.com (as of Sep 2023)



CVE per year – cvedetails.com (as of Sep 2021)



CVSS – Common Vulnerability Scoring System

Provides a quick way to determine the severity of a vulnerability (0-10 score)

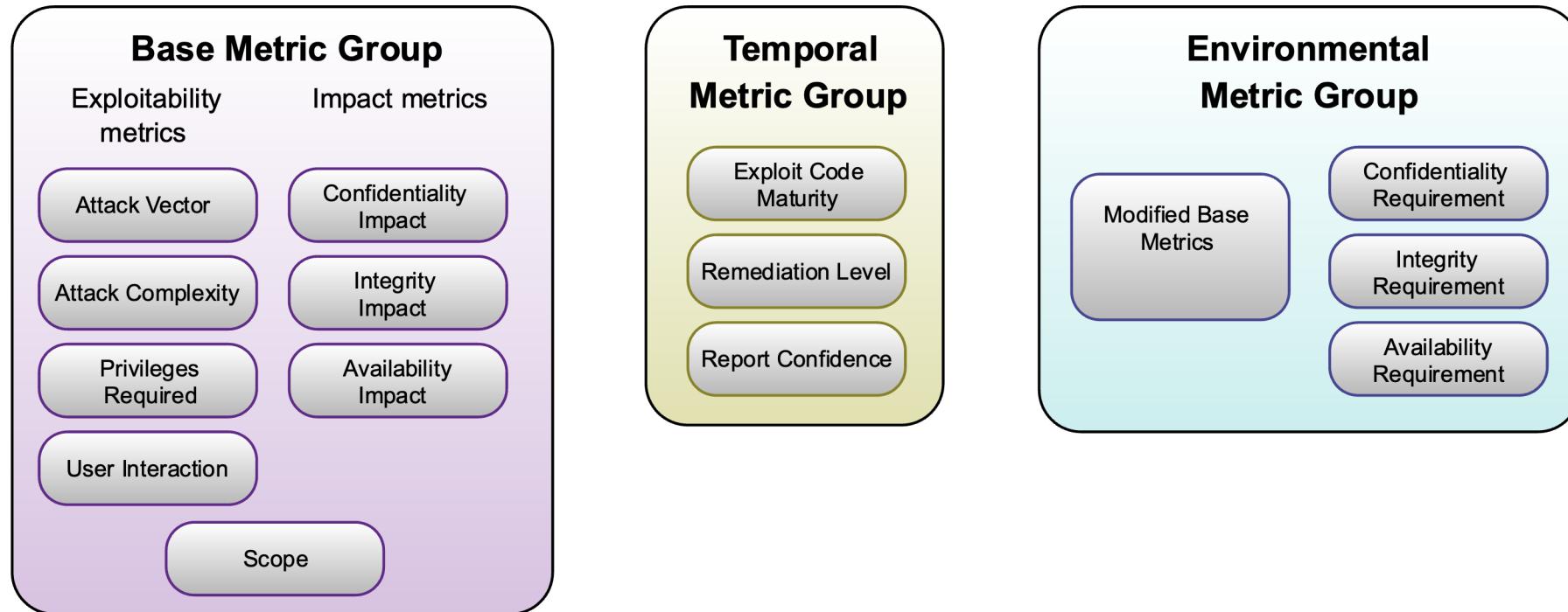
- Helps defenders prioritizing the deployment of mitigations
- Helps attackers selecting the most convenient vulnerability to explore
- Tends to be pessimistic (higher values)

Example: CVSS 3.1/AV:N/AC:L/PR:H/UI:N/S:U/C:L/I:L/A:N

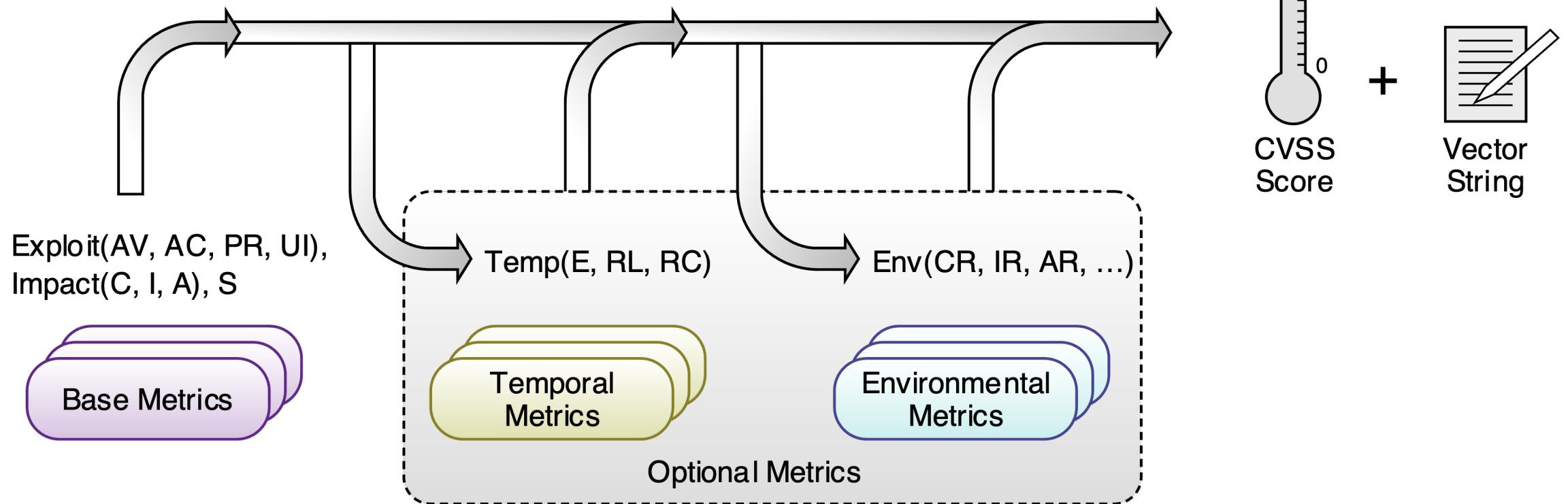
- Final Score: 3.1 (LOW)
- Attack Vector: Network
- Attack Complexity: Low
- Privileges Required: High
- User Interaction: None
- Scope: Unchanged
- Confidentiality: Low
- Integrity: Low
- Exploit Availability: None

CVSS helps decide which vulnerabilities to prioritize.

CVSS – Common Vulnerability Scoring System



CVSS – Common Vulnerability Scoring System



Equations available at: <https://www.first.org/cvss/specification-document>

Calculator available at: <https://www.first.org/cvss/calculator/3.1>

Example: Base Metrics

The Base Score formula depends on sub-formulas for **Impact Sub-Score (ISS)**, **Impact**, and **Exploitability**

$$\text{ISS} = 1 - [(1 - \text{Confidentiality}) \times (1 - \text{Integrity}) \times (1 - \text{Availability})]$$

Impact =	
If Scope is Unchanged	$6.42 \times \text{ISS}$
If Scope is Changed	$7.52 \times (\text{ISS} - 0.029) - 3.25 \times (\text{ISS} - 0.02)^{15}$
Exploitability =	$8.22 \times \text{AttackVector} \times \text{AttackComplexity} \times \text{PrivilegesRequired} \times \text{UserInteraction}$
BaseScore =	
If Impact ≤ 0	$0, \text{ else}$
If Scope is Unchanged	Roundup (Minimum [(Impact + Exploitability), 10])
If Scope is Changed	Roundup (Minimum [1.08 \times (Impact + Exploitability), 10])



Vulnerability Disclosure

How should a research proceed when a vulnerability is found?

If the engagement is private: deliver to contracting entity

- May negotiate the public release the information...

What about other cases?



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

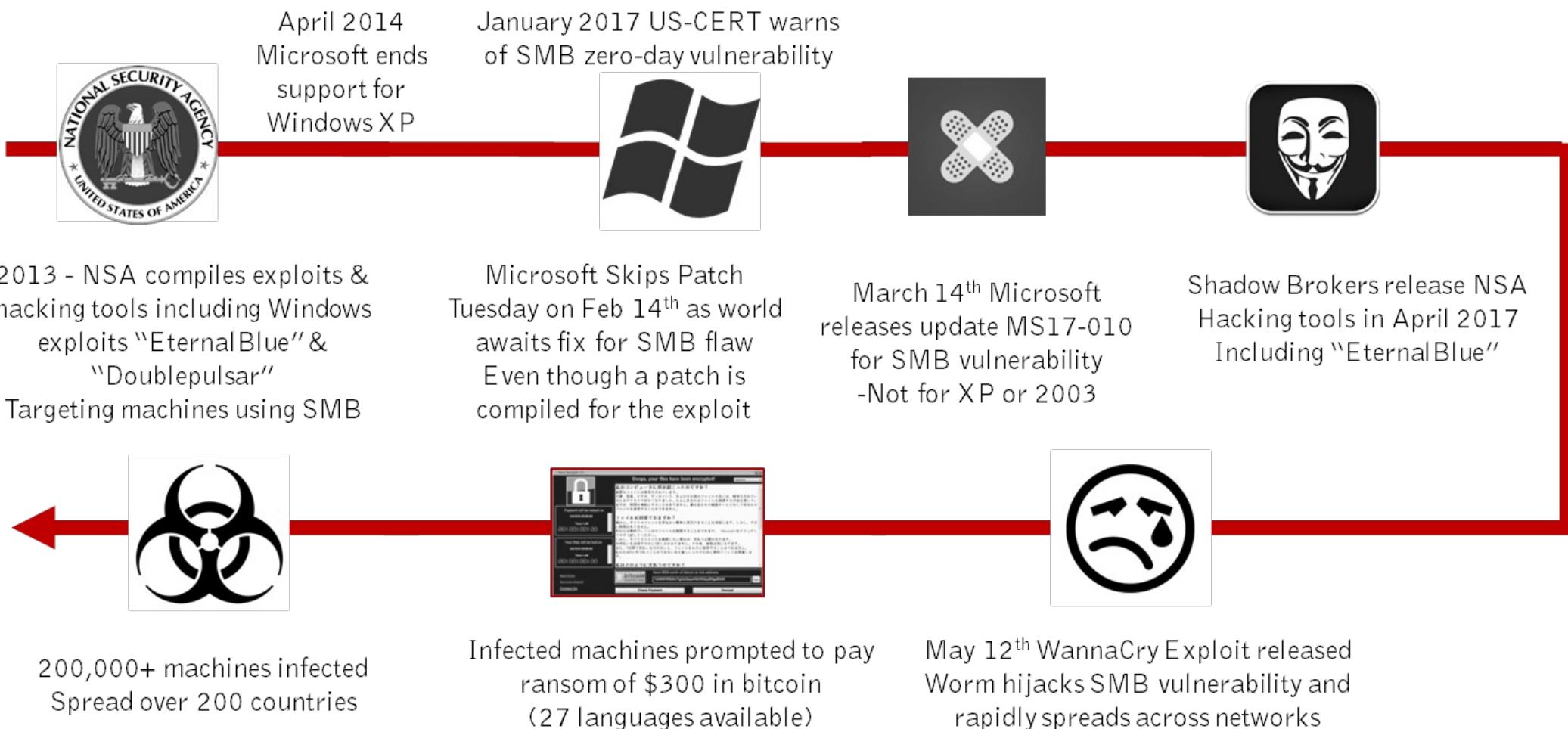
If impact is high, it creates major disruption when publicly known

- Quick adoption in malware and dissemination
 - Remember: Systems take at least one month to be patched



CVE-2017-0144

EternalBlue



Source undetermined



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

→ most common

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

Optional: A website with a fancy name is created for public awareness

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](#).



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

Exercises

This task proposes that a group of 2 students analyze one CVE from the following list and identify:

- what it affected
- what was the vulnerability
- how/when it was discovered
- when was it fixed
- how it was exploited by attackers
- what was the impact of its exploitation
- what was the timeline of major events

CVE-2017-18017 - xt_TCPMSS	CVE-2023-27524 - Superset
CVE-2017-17510 - DLINK Devices	CVE-2023-1748 - Nexx
CVE-2017-5754 - Meltdown	CVE-2023-1424 - Mitsubishi
CVE-2017-5753 - Spectre	CVE-2022-36958 - Print Spooler
CVE-2017-13077 - KRACK	CVE-2021-44228 - Log4j
CVE-2017-0144 - Eternalblue	CVE-2021-26855 - Proxylogon
CVE-2016-10229 - UDP	CVE-2020-9478 - Rubrik CDM
CVE-2015-1538 - Stagefright	CVE-2020-15802 - BLURtooth
CVE-2014-6271 - Shellshock	CVE-2020-1472 - Zerologon
CVE-2014-3566 - Poodle	CVE-2020-0796 - SMBGhost
CVE-2014-0160 - Heartbleed	CVE-2019-17510 - DLINK Devices
CVE-2013-3183 - Ping6 of Death	CVE-2019-15846 - Exim Backslash
CVE-2009-3677 - MSCHAP	CVE-2019-15926 - Linux Out of Bounds
CVE-2008-1447 - Kaminsky DNS	CVE-2017-15846 - Exim Backslash
	CVE-2017-0144 - Eternalblue



Exercises

1. Visit <https://threatpost.com> or <https://www.securityweek.com>
2. For any article:
 - Summarize (or speculate) what went wrong
 - Was it an application problem or an external element?
 - How could it have been avoided? Left right
 - How would your team have reacted?
3. Discuss the point with the class

Both exercises will not be graded

- Only to share some thoughts



