



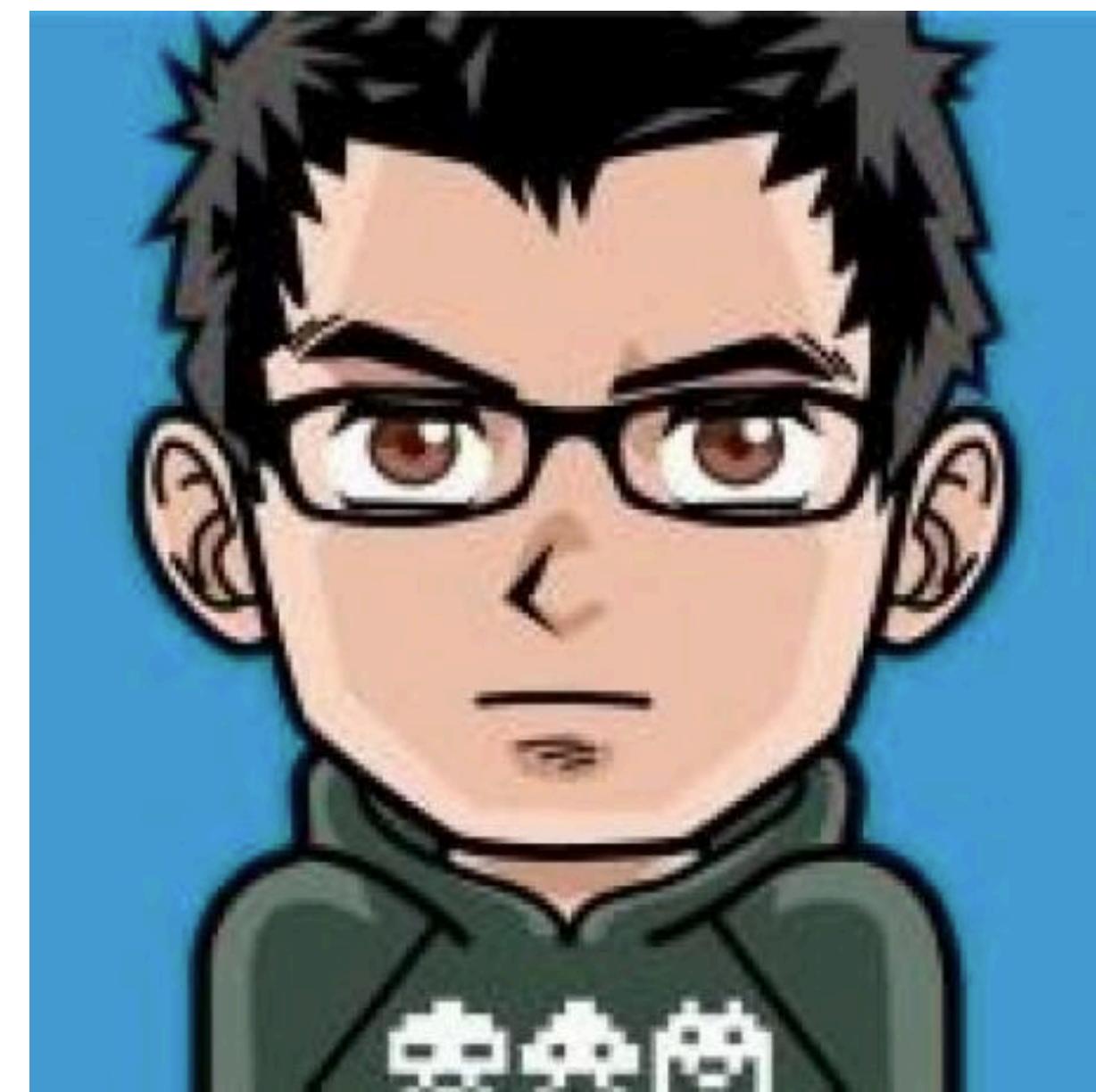
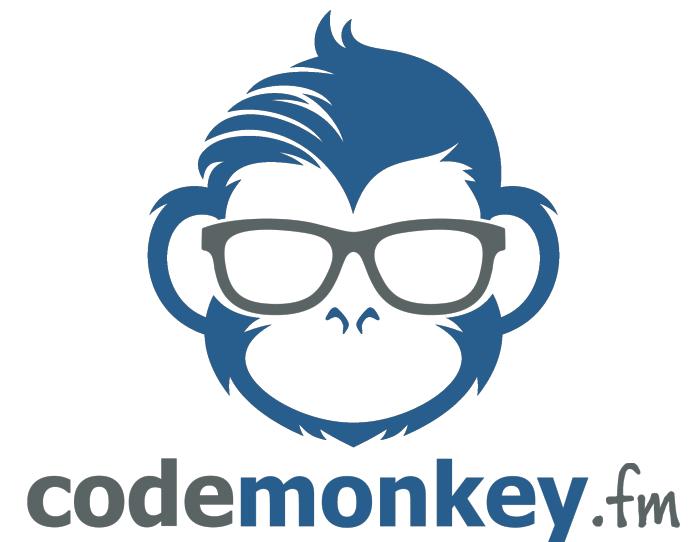
```
/*
 * The default Spring logout behavior redirects a user back to {code}/login?logout, so you will need
 * to change that. The easiest way to do this is by both extending from {link} OAuth2LogoutHandler
 * and annotating your implementation with {@link EnableOAuth2Sso}.
 */
@Configuration
@EnableOAuth2Sso
static class ExampleSecurityConfigurerAdapter extends OAuth2AuthorizationServerConfiguration {
    protected void configure(HttpSecurity http) throws Exception {
        // In this example we allow anonymous access to the root index page
        // this MUST be configured before calling super.configure();
        http.authorizeRequests().antMatchers("/*").permitAll();
        // calling super.configure() before calling http.authorizeRequests() locks everything else out
        super.configure(http);
        // after calling super.configure() you can change the logic here
        http.logout();
    }
}
```

How to Report a Vulnerability: Responsible Disclosure for Developers

Brian Demers
Developer Experience
bdemers@apache.org



Who is this guy?





FRIENDS DON'T LET FRIENDS WRITE AUTH

Topics

- What is a Vulnerability
- What is Responsible Disclosure
- How they are Reported
- What you can do for your Projects

IANAL: I Am Not A Lawyer

TINLA: This Is Not Legal Advice

Quick Example

audible.com/typ/promo?couponValue=1000000.0

The screenshot shows the Audible website interface. At the top, there is a navigation bar with links for "Home", "Library", "Wish List", "Browse", "Listener Page", and "Gift Center". On the right side of the header, there are user account details ("Hi, Brian!"), credit information ("1 Credit Available Buy 3 extra credits"), a coupon balance ("Coupon balance: \$5.00"), and links for "Help" and "Cart". Below the header, a large black banner displays the text "Congratulations! You redeemed a \$1,000,000.00 Coupon. Find your next listen!" in white. To the right of the banner is a search bar with the placeholder "Search for a great book" and a magnifying glass icon.

What is a Vulnerability

vulnerability

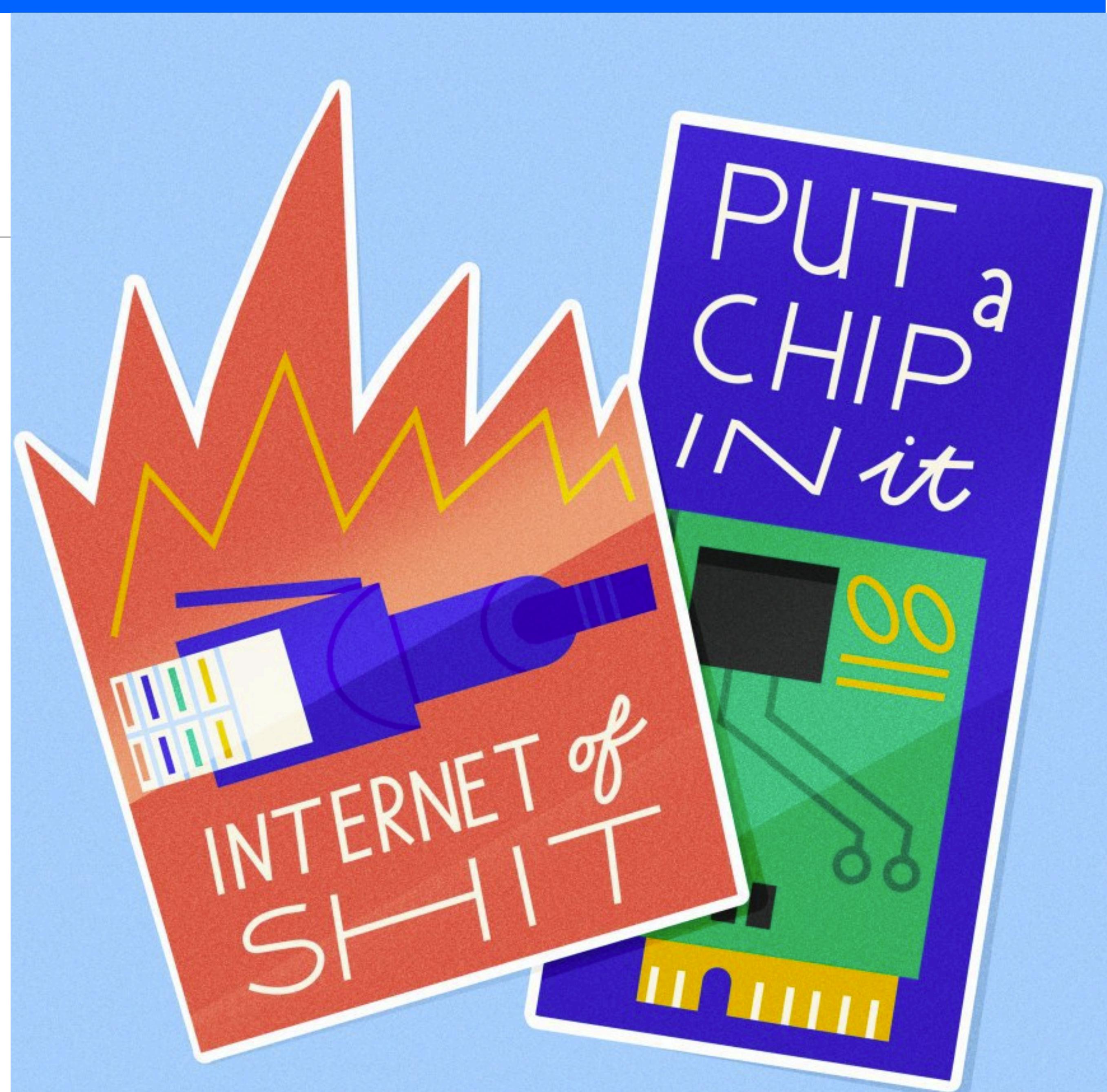
NOUN (vulnerabilities)

- 1 The quality or state of being exposed to the possibility of being attacked or harmed, either physically or emotionally.

Vulnerability (computing)

From Wikipedia, the free encyclopedia

In [computer security](#), a **vulnerability** is a weakness which can be exploited by a [threat actor](#), such as an attacker, to perform unauthorized actions within a computer system.



CVE vs Vulnerability

Common Vulnerabilities and Exposures

- An ID for Vulnerabilities

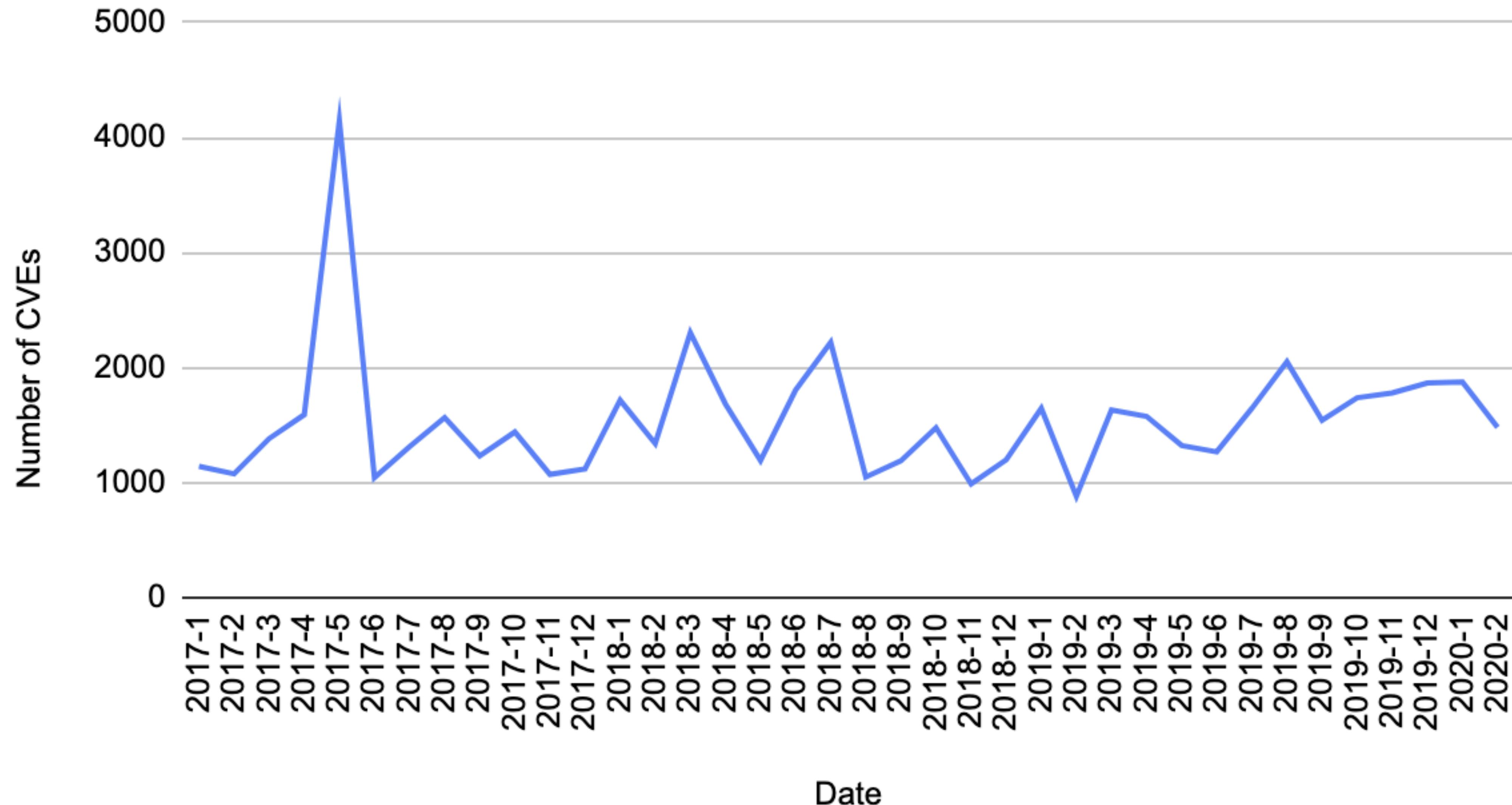
CVE-2018-17793

<year>—<number>

LEAKED LIST OF MAJOR 2018 SECURITY VULNERABILITIES

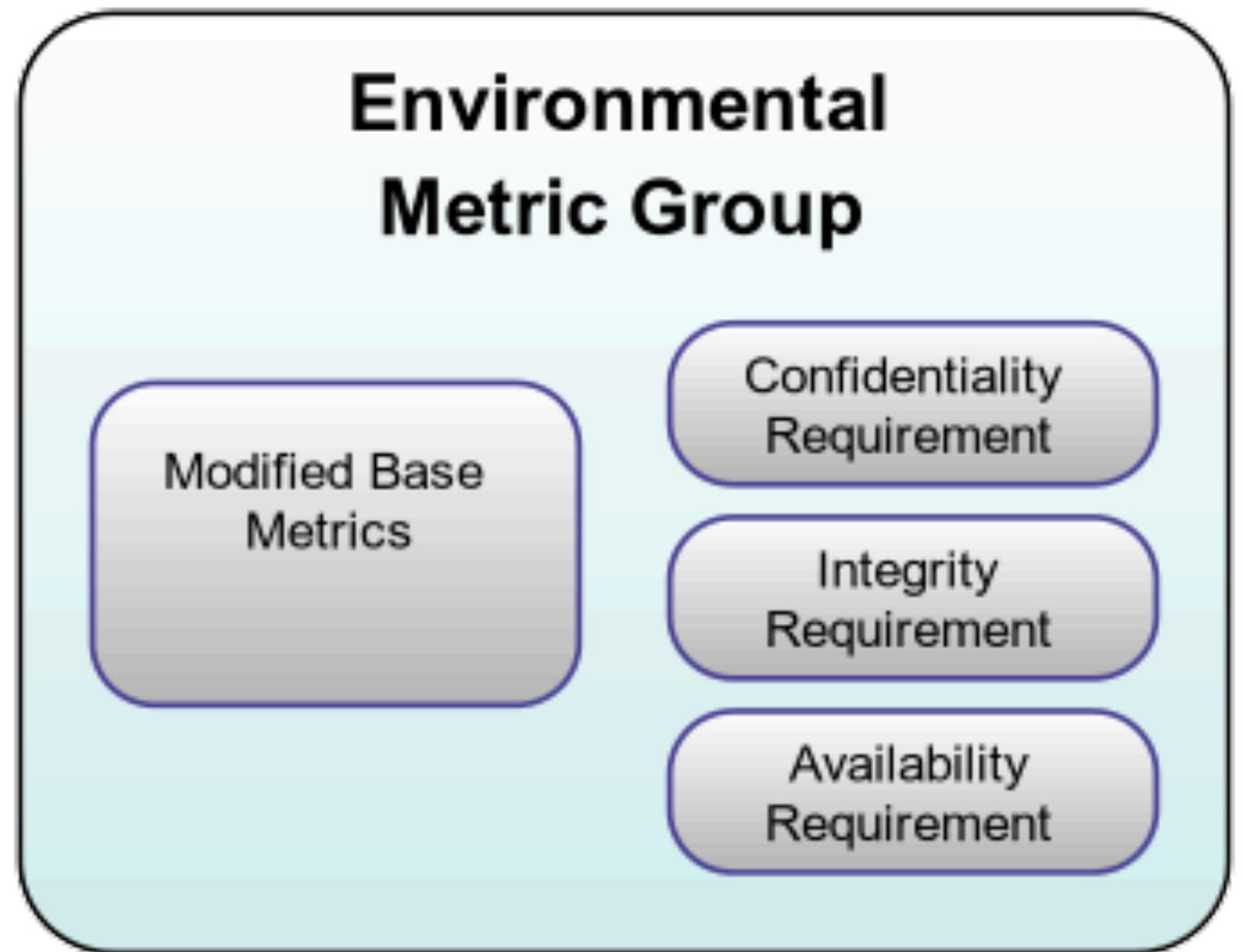
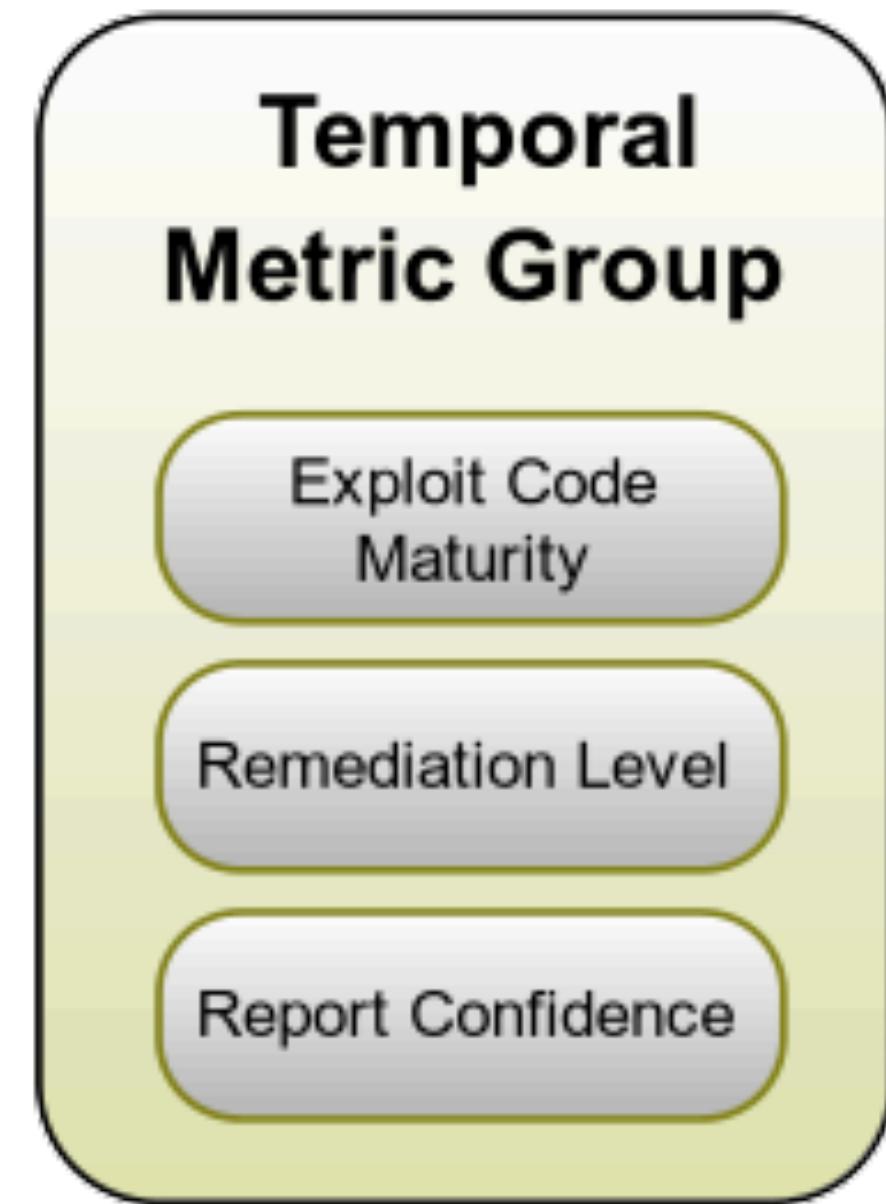
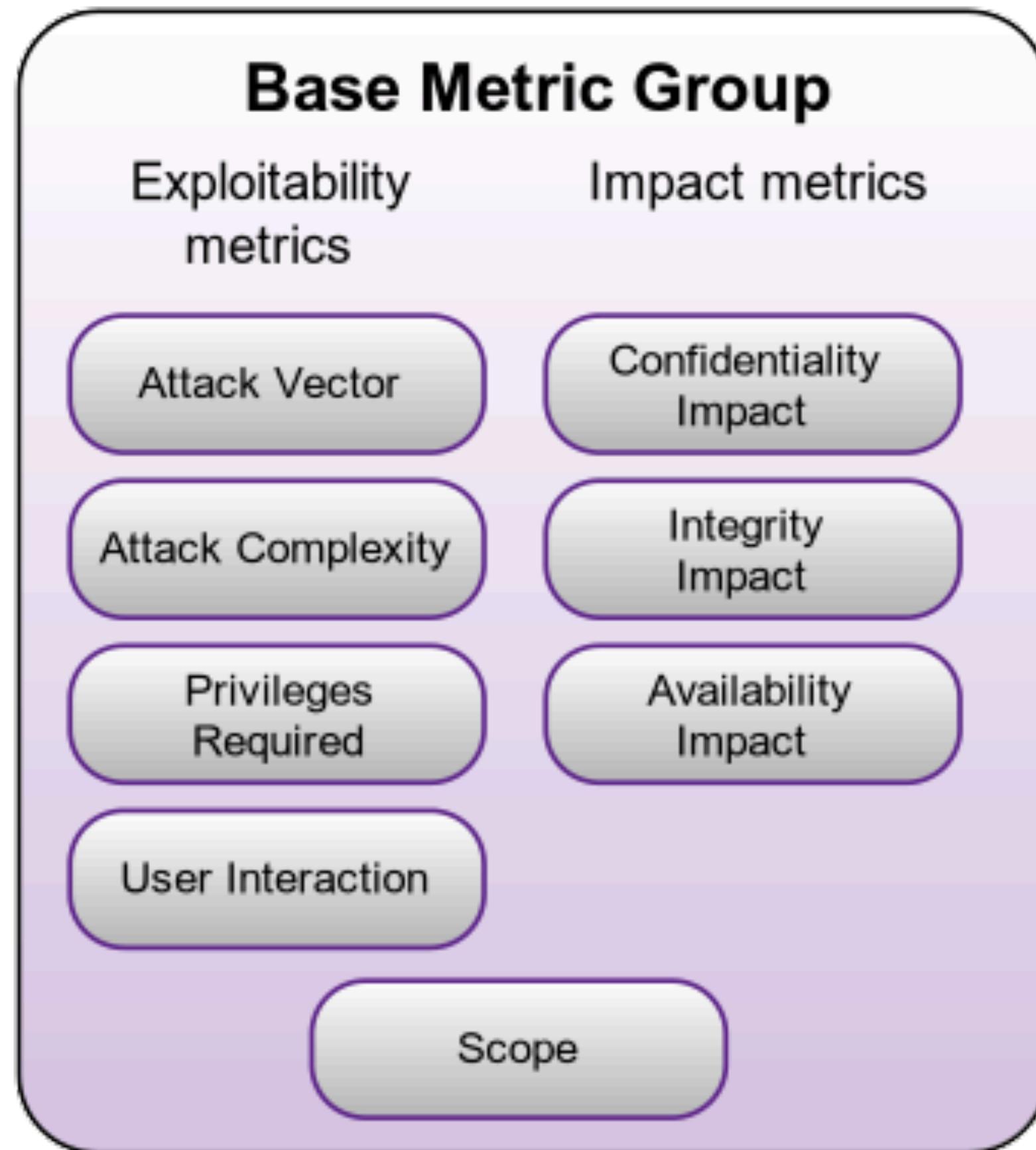
- CVE-2018-????? APPLE PRODUCTS CRASH WHEN DISPLAYING CERTAIN TELUGU OR BENGALI LETTER COMBINATIONS.
- CVE-2018-????? AN ATTACKER CAN USE A TIMING ATTACK TO EXPLOIT A RACE CONDITION IN GARBAGE COLLECTION TO EXTRACT A LIMITED NUMBER OF BITS FROM THE WIKIPEDIA ARTICLE ON CLAUDE SHANNON.
- CVE-2018-????? AT THE CAFE ON THIRD STREET, THE POST-IT NOTE WITH THE WIFI PASSWORD IS VISIBLE FROM THE SIDEWALK.
- CVE-2018-????? A REMOTE ATTACKER CAN INJECT ARBITRARY TEXT INTO PUBLIC-FACING PAGES VIA THE COMMENTS BOX.
- CVE-2018-????? MYSQL SERVER 5.5.45 SECRETLY RUNS TWO PARALLEL DATABASES FOR PEOPLE WHO SAY "S-Q-L" AND "SEQUEL."
- CVE-2018-????? A FLAW IN SOME x86 CPUs COULD ALLOW A ROOT USER TO DE-ESCALATE TO NORMAL ACCOUNT PRIVILEGES.
- CVE-2018-????? APPLE PRODUCTS CATCH FIRE WHEN DISPLAYING EMOJI WITH DIACRITICS.
- CVE-2018-????? AN OVERSIGHT IN THE RULES ALLOWS A DOG TO JOIN A BASKETBALL TEAM.
- CVE-2018-????? HASKELL ISN'T SIDE-EFFECT-FREE AFTER ALL; THE EFFECTS ARE ALL JUST CONCENTRATED IN THIS ONE COMPUTER IN MISSOURI THAT NO ONE'S CHECKED ON IN A WHILE.
- CVE-2018-????? NOBODY REALLY KNOWS HOW HYPERVISORS WORK.
- CVE-2018-????? CRITICAL: UNDER LINUX 3.14.8 ON SYSTEM/390 IN A UTC+14 TIME ZONE, A LOCAL USER COULD POTENTIALLY USE A BUFFER OVERFLOW TO CHANGE ANOTHER USER'S DEFAULT SYSTEM CLOCK FROM 12-HOUR TO 24-HOUR.
- CVE-2018-????? x86 HAS WAY TOO MANY INSTRUCTIONS.
- CVE-2018-????? NUMPY 1.8.0 CAN FACTOR PRIMES IN O(LOG N) TIME AND MUST BE QUIETLY DEPRECATED BEFORE ANYONE NOTICES.
- CVE-2018-????? APPLE PRODUCTS GRANT REMOTE ACCESS IF YOU SEND THEM WORDS THAT BREAK THE "I BEFORE E" RULE.
- CVE-2018-????? SKYLAKE x86 CHIPS CAN BE PRIED FROM THEIR SOCKETS USING CERTAIN FLATHEAD SCREWDRIVERS.
- CVE-2018-????? APPARENTLY LINUS TORVALDS CAN BE BRIBED PRETTY EASILY.
- CVE-2018-????? AN ATTACKER CAN EXECUTE MALICIOUS CODE ON THEIR OWN MACHINE AND NO ONE CAN STOP THEM.
- CVE-2018-????? APPLE PRODUCTS EXECUTE ANY CODE PRINTED OVER A PHOTO OF A DOG WITH A SADDLE AND A BABY RIDING IT.
- CVE-2018-????? UNDER RARE CIRCUMSTANCES, A FLAW IN SOME VERSIONS OF WINDOWS COULD ALLOW FLASH TO BE INSTALLED.
- CVE-2018-????? TURNS OUT THE CLOUD IS JUST OTHER PEOPLE'S COMPUTERS.
- CVE-2018-????? A FLAW IN MITRE'S CVE DATABASE ALLOWS ARBITRARY CODE INSERTION. [~CLICK HERE FOR CHEAP VIAGRA~]

Number of CVEs / Month



(Data from nvd.nist.gov)

CVSS Score



nvd.nist.gov/vuln-metrics/cvss/v3-calculator

Common Weakness Enumeration (CWE)

CWE-250: Execution with Unnecessary Privileges

CWE-94: Improper Control of Generation of Code ('Code Injection')

Common Platform Enumeration (CPE)

cpe:2.3:o:microsoft:windows_xp:-:sp3:*:***:x86:***

cpe:2.3:a:apache:struts:*:***:***:***:***:*****

org.apache.struts:struts2-core:2.5.0

CVEs are Bad?



Apache Tomcat® - Reporting Secu X +

tomcat.apache.org/security.html Private

Apache Tomcat®

 SUPPORT THE APACHE SOFTWARE FOUNDATION

Search... GO

 APACHE EVENTS LEARN MORE

[Save the date!](#)

Apache Tomcat

- Home
- Taglibs
- Maven Plugin

Download

- Which version?
- Tomcat 10
- Tomcat 9
- Tomcat 8
- Tomcat 7
- Tomcat Connectors
- Tomcat Native
- Taglibs
- Archives

Documentation

- Tomcat 10.0
- Tomcat 9.0
- Tomcat 8.5
- Tomcat 7.0
- Tomcat Connectors
- Tomcat Native
- Wiki
- Migration Guide
- Presentations

Problems?

- Security Reports
- Find help
- FAQ

Security Updates

Please note that, except in rare circumstances, binary patches are not produced for individual vulnerabilities. To obtain the binary fix for a particular vulnerability you should upgrade to an Apache Tomcat version where that vulnerability has been fixed.

Source patches, usually in the form of references to commits, may be provided in either in a vulnerability announcement and/or the vulnerability details listed on these pages. These source patches may be used by users wishing to build their own local version of Tomcat with just that security patch rather than upgrade. Please note that an exercise is currently underway to add links to the commits for all the vulnerabilities.

Lists of security problems fixed in released versions of Apache Tomcat are available:

- [Apache Tomcat 10.x Security Vulnerabilities](#)
- [Apache Tomcat 9.x Security Vulnerabilities](#)
- [Apache Tomcat 8.x Security Vulnerabilities](#)
- [Apache Tomcat 7.x Security Vulnerabilities](#)
- [Apache Tomcat JK Connectors Security Vulnerabilities](#)
- [Apache Tomcat APR/native Connector Security Vulnerabilities](#)
- [Apache Taglibs Security Vulnerabilities](#)

Lists of security problems fixed in versions of Apache Tomcat that may be downloaded are:

- [Apache Tomcat 6.x Security Vulnerabilities](#)
- [Apache Tomcat 5.x Security Vulnerabilities](#)
- [Apache Tomcat 4.x Security Vulnerabilities](#)
- [Apache Tomcat 3.x Security Vulnerabilities](#)

Reporting New Security Problems with Apache Tomcat

The Apache Software Foundation takes a very active stance in eliminating security problems from our software. We strongly encourage folks to report such problems to our private security mailing list.

Please note that the security mailing list should only be used for reporting undiscovered security problems. We cannot accept regular bug reports or other types of reports. If you have a problem that you believe relates to an undisclosed security problem in the Apache Tomcat source code we would appreciate your help in fixing it.

If you need to report a bug that isn't an undisclosed security vulnerability, please use the Apache JIRA issue tracking system.

 Common Vulnerabilities and Exposures

CVE List CNAs WGs Board About News & Blog NVD Go to for: CVSS Scores CPE Info Advanced Search TOTAL CVE Entries: 131433

Search CVE List Download CVE Data Feeds Request CVE IDs Update a CVE Entry

HOME > CVE > SEARCH RESULTS

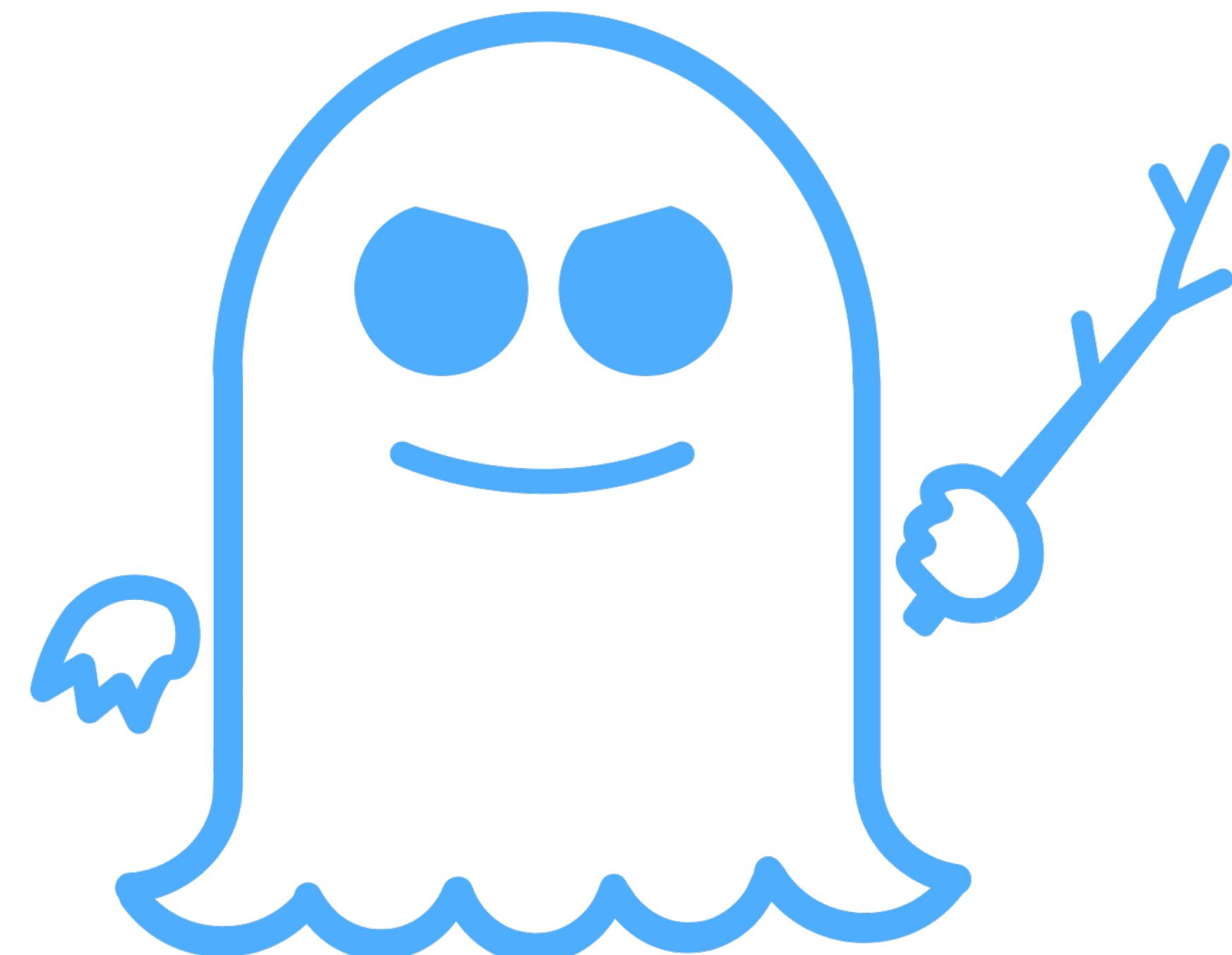
Search Results

There are 198 CVE entries that match your search.

Name	Description
CVE-2020-1938	When using the Apache JServ Protocol (AJP), care must be taken when trusting incoming connections to Apache Tomcat. Tomcat treats AJP connections as having higher trust than, for example, a similar HTTP connection. If such connections are available to an attacker, they can be exploited in ways that may be surprising. In Apache Tomcat 9.0.0.M1 to 9.0.0.30, 8.5.0 to 8.5.50 and 7.0.0 to 7.0.99, Tomcat shipped with an AJP Connector enabled by default that listened on all configured IP addresses. It was expected (and recommended in the security guide) that this Connector would be disabled if not required. This vulnerability report identified a mechanism that allowed: - returning arbitrary files from anywhere in the web application - processing any file in the web application as a JSP. Further, if the web application allowed file upload and stored those files within the web application (or the attacker was able to control the content of the web application by some other means) then this, along with the ability to process a file as a JSP, made remote code execution possible. It is important to note that mitigation is only required if an AJP port is accessible to untrusted users. Users wishing to take a defence-in-depth approach and block the vector that permits returning arbitrary files and execution as JSP may upgrade to Apache Tomcat 9.0.31, 8.5.51 or 7.0.100 or later. A number of changes were made to the default AJP Connector configuration in 9.0.31 to harden the default configuration. It is likely that users upgrading to 9.0.31, 8.5.51 or 7.0.100 or later will need to make small changes to their configurations.
CVE-2020-1935	In Apache Tomcat 9.0.0.M1 to 9.0.30, 8.5.0 to 8.5.50 and 7.0.0 to 7.0.99 the HTTP header parsing code used an approach to end-of-line parsing that allowed some invalid HTTP headers to be parsed as valid. This led to a possibility of HTTP Request Smuggling if Tomcat was located behind a reverse proxy that incorrectly handled the invalid Transfer-Encoding header in a particular manner. Such a reverse proxy is considered unlikely.
CVE-2019-17569	The refactoring present in Apache Tomcat 9.0.28 to 9.0.30, 8.5.48 to 8.5.50 and 7.0.98 to 7.0.99 introduced a regression. The result of the regression was that invalid Transfer-Encoding headers were incorrectly processed leading to a possibility of HTTP Request Smuggling if Tomcat was located behind a reverse proxy that incorrectly handled the invalid Transfer-Encoding header in a particular manner. Such a reverse proxy is considered unlikely.
CVE-2019-17563	When using FORM authentication with Apache Tomcat 9.0.0.M1 to 9.0.29, 8.5.0 to 8.5.49 and 7.0.0 to 7.0.98 there was a narrow window where an attacker could perform a session fixation attack. The window was considered too narrow for an exploit to be practical but, erring on the side of caution, this issue has been treated as a security vulnerability.
CVE-2019-12418	When Apache Tomcat 9.0.0.M1 to 9.0.28, 8.5.0 to 8.5.47, 7.0.0 and 7.0.97 is configured with the JMX Remote Lifecycle Listener, a local attacker without access to the Tomcat process or configuration files is able to manipulate the RMI registry to perform a man-in-the-middle attack.

- <https://cve.mitre.org/>
- <https://nvd.nist.gov/>

The Bad...



SPECTRE



Responsible Disclosure

- Give vendor time to fix vulnerability before telling public

Full Disclosure

- Tell public ASAP



EMBARGO

How long to wait?

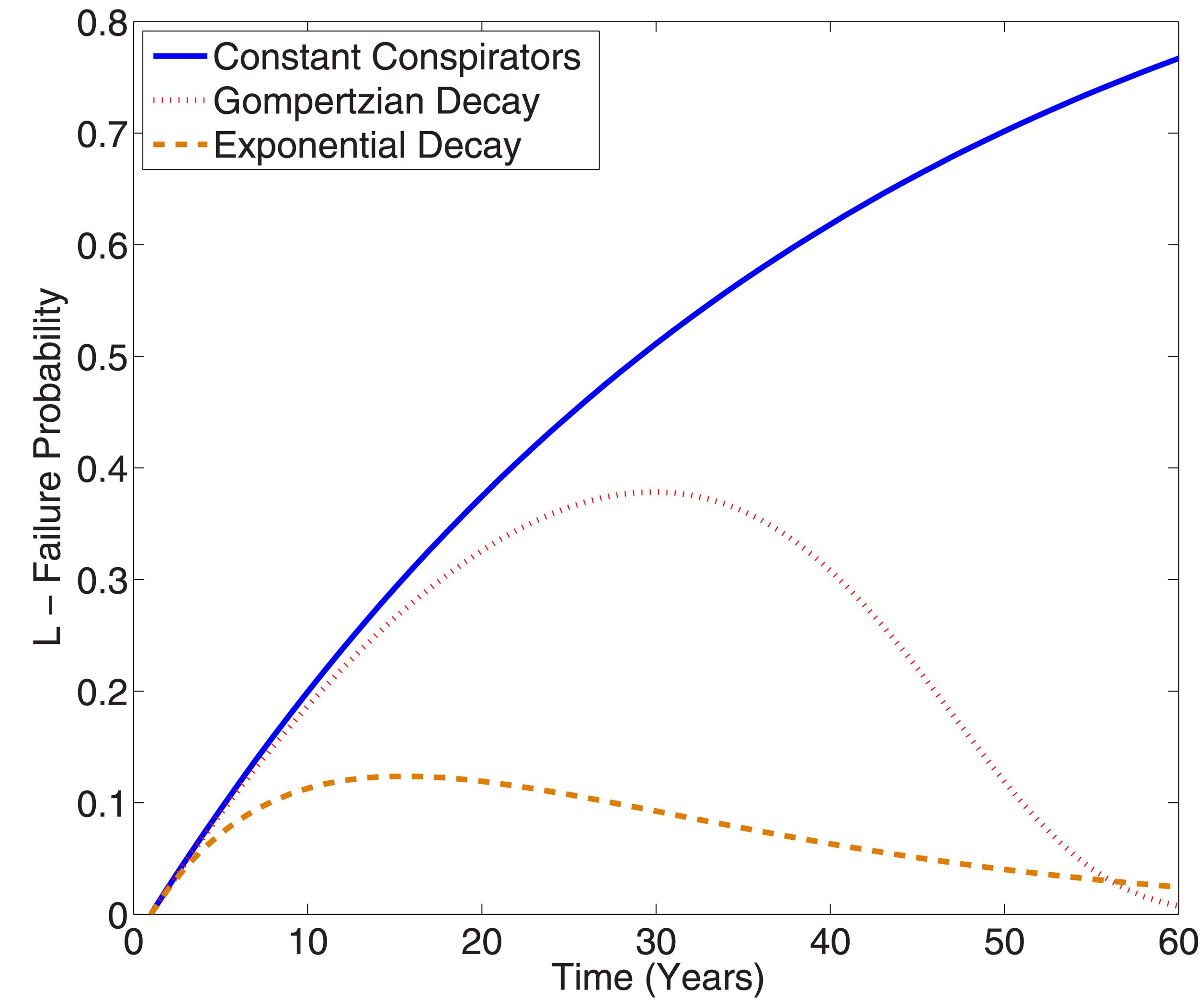
- Google Project Zero - 90 days
- Linux Kernel - 2 weeks
- HackerOne - 30 days
- CERT - 45 days



On the Viability of Conspiratorial Beliefs

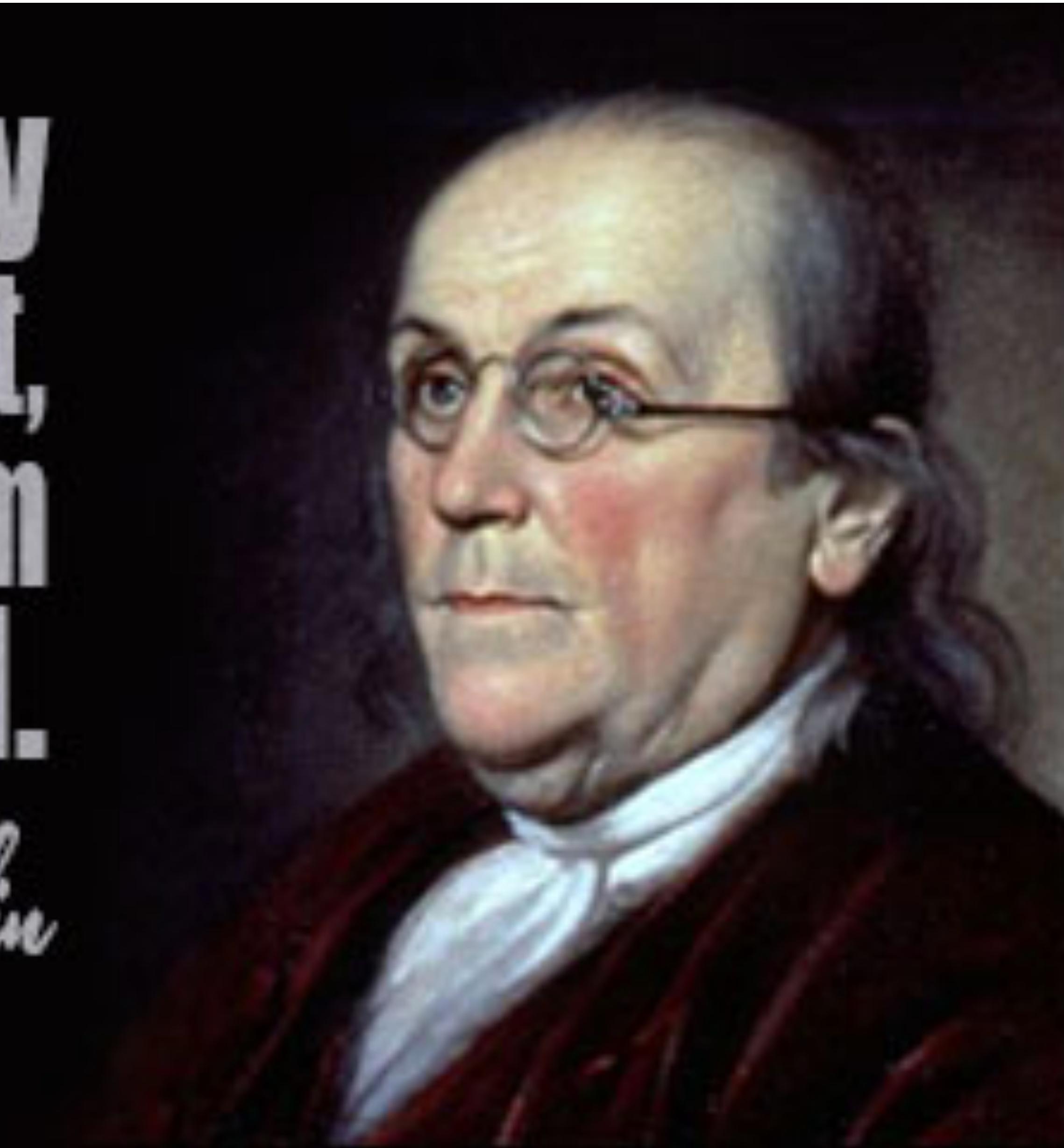
David Robert Grimes

TL;DR - The more people that know
the more likely it will get out.



**Three may
keep a secret,
if two of them
are dead.**

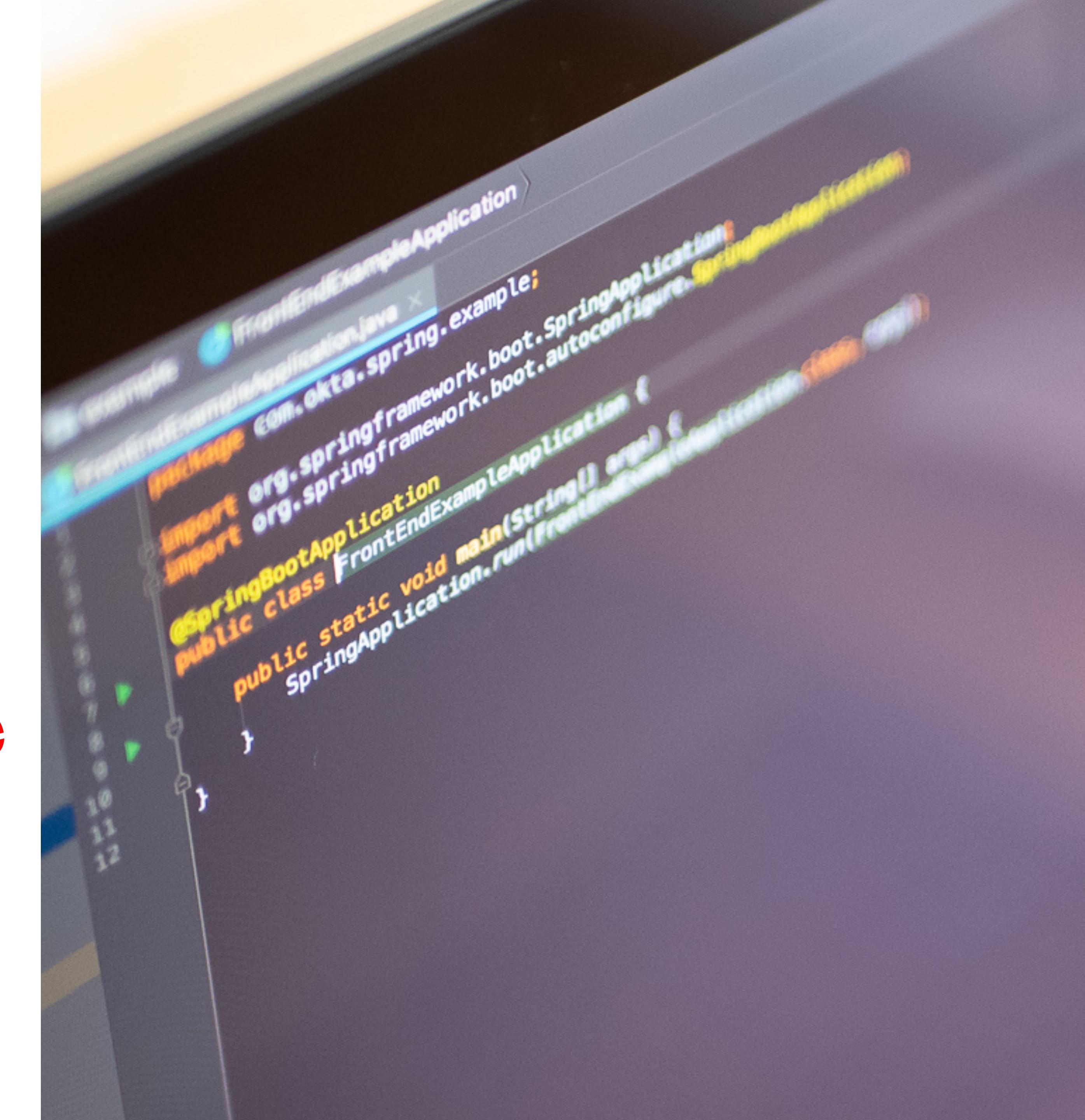
--Benjamin Franklin



How are Vulnerabilities Reported?

- Report the issue to the project
 - Wait for response
- Vendor releases patch/fix
 - Wait for customers to apply fix
- Disclose issue publicly

Report, Fix, Disclose



```
FrontEndExampleApplication.java
package com.okta.spring.example;
import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
@SpringBootApplication
public class FrontEndExampleApplication {
    public static void main(String[] args) {
        SpringApplication.run(FrontEndExampleApplication.class, args);
    }
}
```

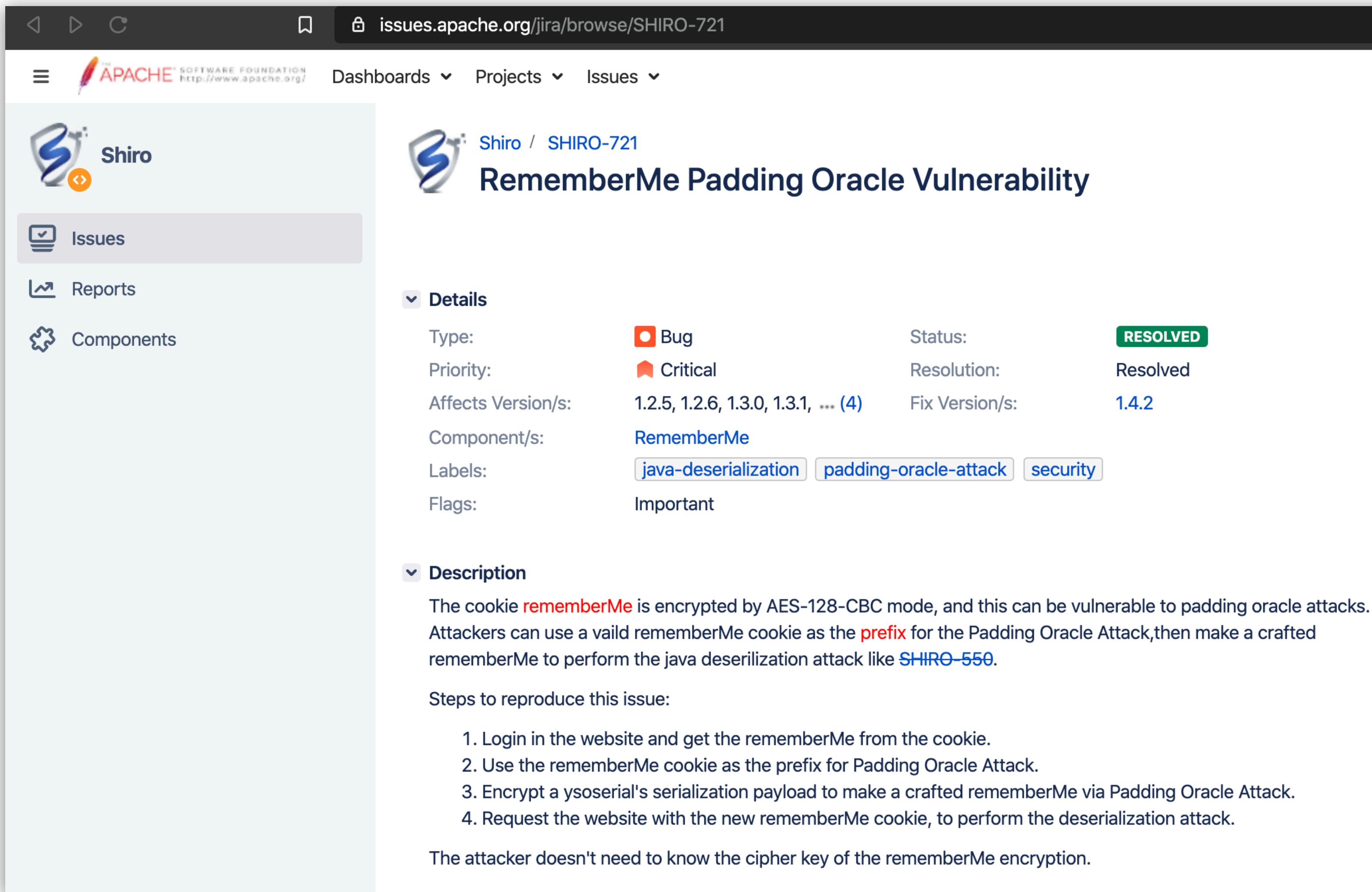
Report (Privately)

- **NOT** on StackOverflow
- **NOT** on an Email list
- **NOT** on an open forum (Slack)
- Look for a security mailing list
- Check Bugcrowd or HackerOne
- If you are worried use an anonymous email account



www.veryfunnypics.eu

Don't use a public bug tracker



The screenshot shows a JIRA issue page for Apache Shiro. The URL in the address bar is issues.apache.org/jira/browse/SHIRO-721. The page title is "Shiro / SHIRO-721 RememberMe Padding Oracle Vulnerability". The left sidebar has links for "Issues", "Reports", and "Components". The main content area shows the issue details:

Type:	Bug	Status:	RESOLVED
Priority:	Critical	Resolution:	Resolved
Affects Version/s:	1.2.5, 1.2.6, 1.3.0, 1.3.1, ... (4)	Fix Version/s:	1.4.2
Component/s:	RememberMe		
Labels:	java-deserialization, padding-oracle-attack, security		
Flags:	Important		

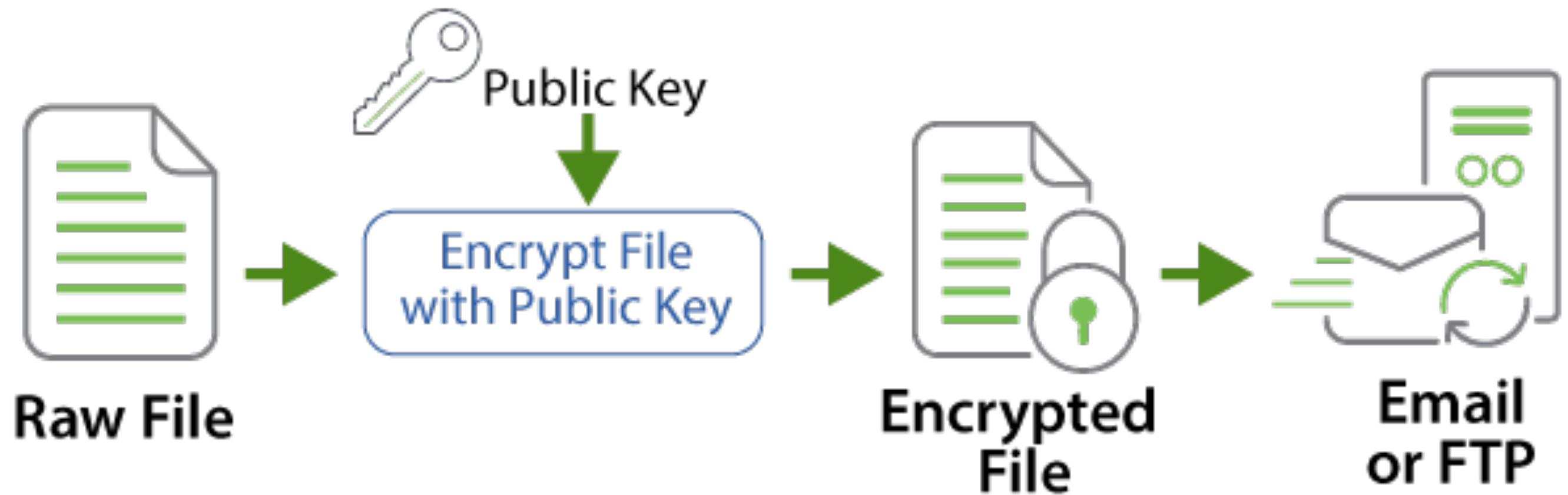
Description
The cookie `rememberMe` is encrypted by AES-128-CBC mode, and this can be vulnerable to padding oracle attacks. Attackers can use a valid rememberMe cookie as the `prefix` for the Padding Oracle Attack, then make a crafted rememberMe to perform the java deserialization attack like [SHIRO-550](#).

Steps to reproduce this issue:

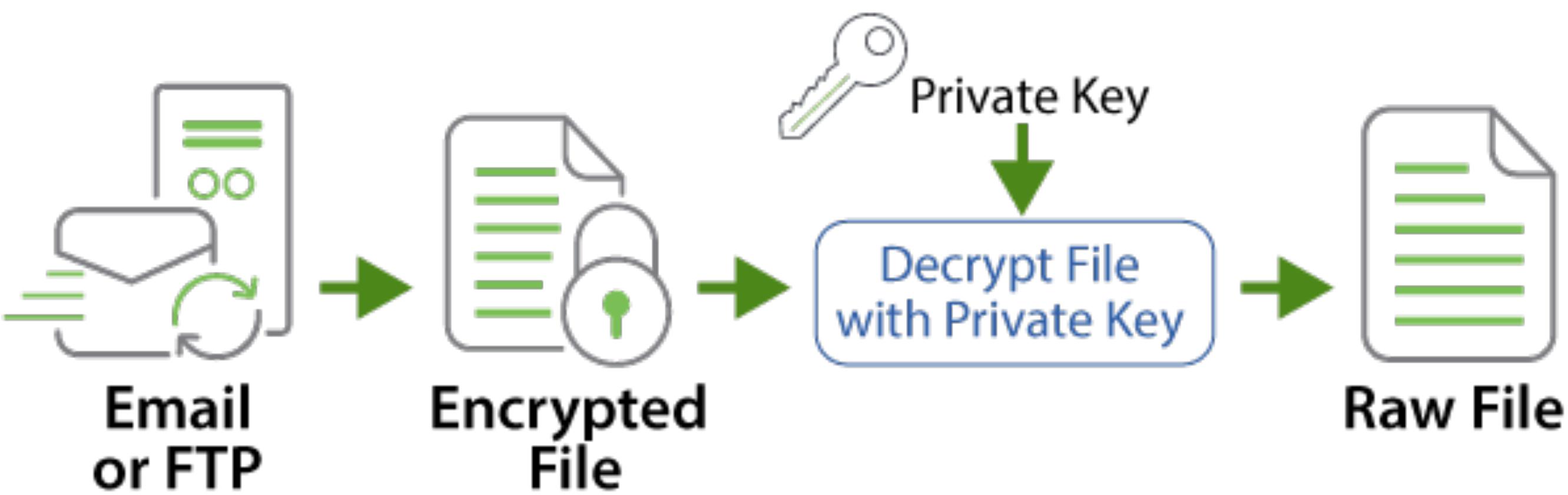
1. Login in the website and get the rememberMe from the cookie.
2. Use the rememberMe cookie as the prefix for Padding Oracle Attack.
3. Encrypt a ysoserial's serialization payload to make a crafted rememberMe via Padding Oracle Attack.
4. Request the website with the new rememberMe cookie, to perform the deserialization attack.

The attacker doesn't need to know the cipher key of the rememberMe encryption.

Encryption Process



Decryption Process



```
$ gpg --recipient <key-id/email>\n--encrypt <input-file>
```

Encrypt Content If Needed

The screenshot shows a web browser with several tabs open:

- okta.com/vulnerability-reporting-policy/pgp-key/
- amazon.com/gp/help/customer/display.html?nodeId=201909050
- amazon.com
- apache.org/security/

The main content area displays information about the Apache Security Team and reporting vulnerabilities:

THE APACHE SECURITY TEAM

The Apache Security Team exists to provide help and advice to Apache projects on security issues and to provide co-ordination of the handling of security vulnerabilities.

REPORTING A VULNERABILITY

We strongly encourage the reporting of potential security vulnerabilities to one of our private security mailing lists first, before disclosing them in a public forum.

A list of security contacts for Apache projects is available. If you can't find a project specific security e-mail address and you have an undisclosed security vulnerability to report then please use the general security address below.

Please note that the security contacts should only be used for reporting undisclosed security vulnerabilities in Apache projects and managing the process of fixing such vulnerabilities. We cannot accept regular bug reports or other security related queries at these addresses. All mail sent to these addresses that does not relate to an undisclosed security problem in an Apache project will be ignored.

Also note that the security team handles vulnerabilities in Apache projects, not running ASF services. All reports of vulnerabilities in ASF services should be sent to root@apache.org only.

The general security mailing list address is: security@apache.org. This is a private mailing list.

Please send one plain-text email for each vulnerability you are reporting. We may ask you to resubmit your report if you send it as an image, movie, HTML, or PDF attachment when it could just as easily be described with plain text.

Encrypted submissions are not required or preferred as it will take us much longer to respond to these reports. There is no team key for security@apache.org instead you can use the OpenPGP keys of the following subset of members of the Apache Security Team. Note that this is not a complete list of Apache Security Team members and that you should not contact these members individually about security issues.

- Mark Cox - 5B25 45DA B219 95F4 088C EFAA 36CE E4DE B00C FE33 - pgp.mit.edu
- Bill Rowe - B1B9 6F45 DFBD CCF9 7401 9235 193F 180A B55D 9977 - pgp.mit.edu
- Mark Thomas - A9C5 DF4D 22E9 9998 D987 5A51 10C0 1C5A 2F60 59E7 - pgp.mit.edu
- Yann Ylavic - 8935 9267 45E1 CE7E 3ED7 48F6 EC99 EE26 7EB5 F61A - pgp.mit.edu

Fix

- It's up to the project to fix the issue
- If it's an open source project get involved!
- The project publishes a release/patch/fix publicly
- The project should contact you with details and a timeline of the fix

Disclose

- After fix is released you can disclose the issue
- Blog about it
- Tell your friends you are a security researcher now
- Or not (some companies reward \$\$ for not talking about it)

The ASF Process



- A detailed 16 step process
- apache.org/security/committers.html

VULNERABILITY HANDLING

A typical process for handling a new security vulnerability is as follows. Projects that wish to use other processes MAY do so, but MUST clearly and publicly document their process and have security@ review it ahead of time.

Note: No information should be made public about the vulnerability until it is formally announced at the end of this process. That means, for example that a Jira issue must NOT be created to track the issue since that will make the issue public. Also the messages associated with any commits should not make ANY reference to the security nature of the commit.

1. The person discovering the issue, the reporter, reports the vulnerability privately to security@project.apache.org or to security@apache.org
2. Messages that do not relate to the reporting or managing of an undisclosed security vulnerability in Apache software are ignored and no further action is required.
3. If reported to security@apache.org, the security team will forward the report (without acknowledging it) to the project's security list or, if the project does not have a security list, to the project's private (PMC) mailing list.
4. The project team sends an e-mail to the original reporter to acknowledge the report. This e-mail must be cc'd to security@project.apache.org if it exists, or security@apache.org otherwise.
5. The project team investigates report and either rejects it or accepts it.
6. If the report is rejected, the project team writes to the reporter to explain why. This e-mail must be cc'd to security@project.apache.org if it exists, or security@apache.org otherwise.
7. If the report is accepted, the project team writes to reporter to let them know it is accepted and that they are working on a fix.
8. The project team requests a CVE number from security@apache.org by sending an e-mail with the subject "CVE request for..." and providing a short (one line) description of the vulnerability. [Guidance](#) is available to determine if a report requires multiple CVEs or if multiple reports should be merged under a single CVE.
9. The project team agrees the fix on their private list.
10. The project team provides the reporter with a copy of the fix and a draft vulnerability announcement for comment.
11. The project team agrees the fix, the announcement and the release schedule with the reporter. For an example of an announcement see [Tomcat's announcement of CVE-2008-2370](#). The level of detail to include in the report is a matter of judgement. Generally, reports should contain enough information to enable people to assess the risk associated with the vulnerability for their system and no more. Steps to reproduce the vulnerability are not normally included.
12. The project team commits the fix. No reference should be made to the commit being related to a security vulnerability.



ASK WHAT YOU CAN DO

FOR YOUR PROJECT

imgflip.com

Search or jump to... / Pull requests Issues Marketplace Explore

spring-projects / spring-security Used by 89.7k Watch 414 Star 4.4k

Create a GitHub Issues Template

Code Issues 862 Pull requests 37 Actions Projects 0 Wiki Security Insights

Helpful resources Contributing GitHub Community Guide

spring-security/.github ISSUE_TEMPLATE.md Cancel

Looks like this file is an issue template. Need help? [Learn more](#).

Edit file Preview changes Spaces 2

```
1 <!--
2 For Security Vulnerabilities, please use https://pivotal.io/security#reporting
3 -->
4
5 ### Summary
6
7 <!--
8 Please provide a high level summary of the issue you are having
9 -->
10
11 ### Actual Behavior
12
13 <!--
14 Please describe step by step the behavior you are observing
15 -->
16
17 ### Expected Behavior
18
19 <!--
20 Please describe step by step the behavior you expect
21 -->
```

Title Related Issues Beta Try it.

Write Preview AA B i “ < > @

<!--
For Security Vulnerabilities, please use <https://pivotal.io/security#reporting>
-->

Summary

<!--
Please provide a high level summary of the issue you are having
-->

Actual Behavior

<!--
Please describe step by step the behavior you are observing
-->

Expected Behavior

<!--
Please describe step by step the behavior you expect
-->

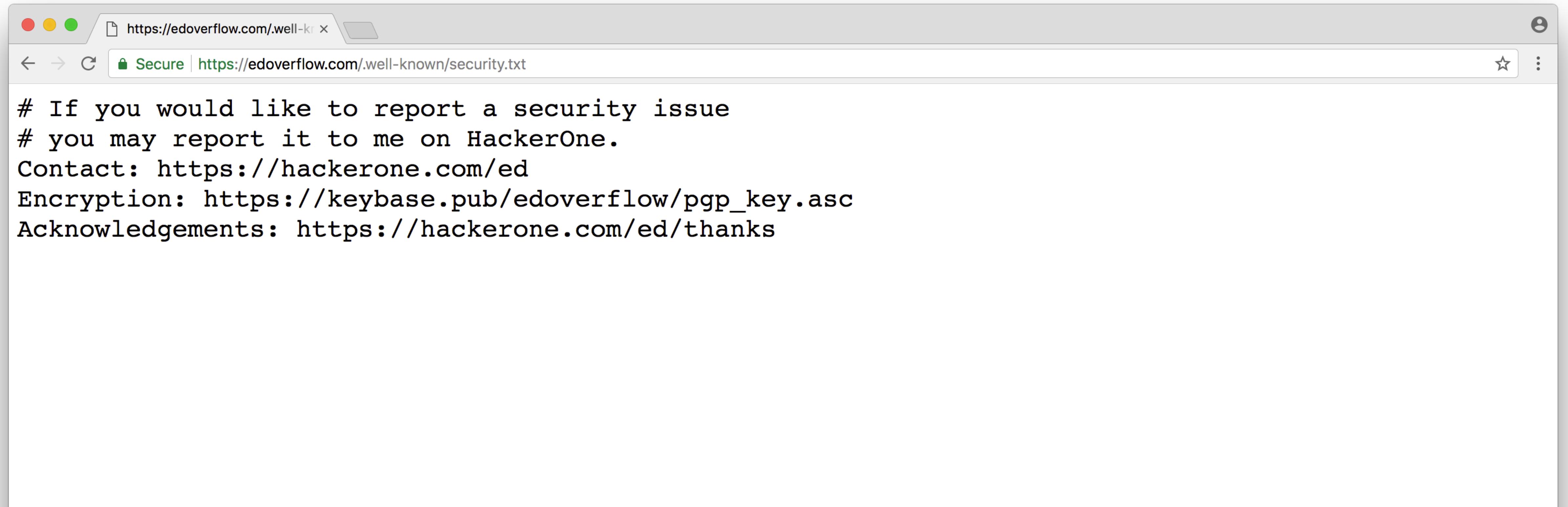
Configuration

Attach files by dragging & dropping, selecting or pasting them.

M Styling with Markdown is supported Submit new issue

securitytxt.org

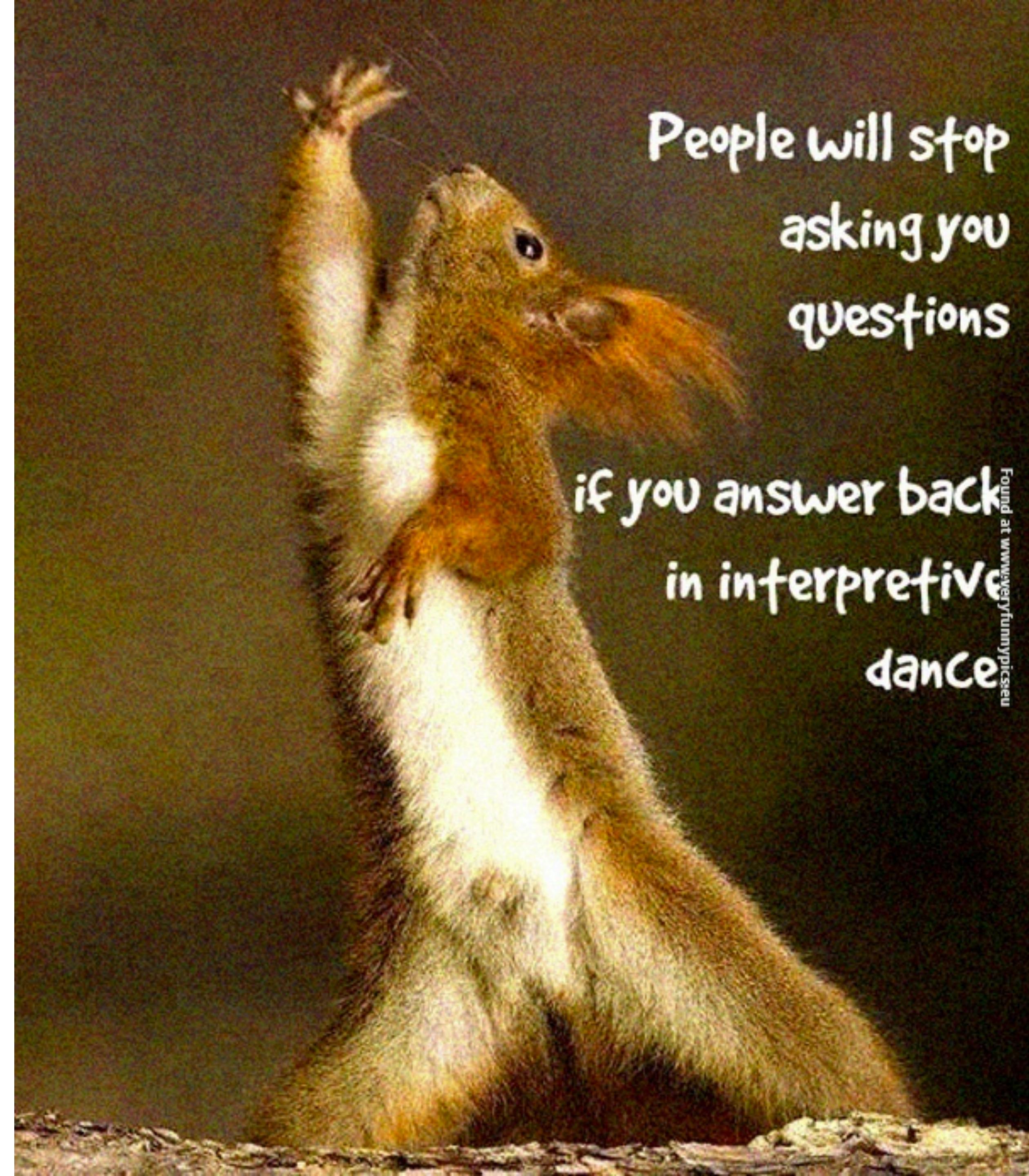
Add a .well-known/security.txt



Questions?

 @BrianDemers

<https://developer.okta.com>



Attribution

- "xkcd 1938, 1957" are licensed under CC BY-NC 2.5
- Internet Of Shit sticker image from: <https://twitter.com/internetofshit>
- <https://intezer.com/wp-content/uploads/2017/08/GoodBAd-1000x475.b197b0.webp>
- Intel -insider trading image: <https://i.kym-cdn.com/photos/images/newsfeed/001/329/141/44f.png>
- Three people secret image: http://www.notable-quotes.com/f/benjamin_franklin_quote_2.jpg
- Secret stamp: cc-by-sa Willscrlt: https://commons.wikimedia.org/wiki/File:Top_secret.png
- Questions image: <https://veryfunnypics.eu/wp-content/uploads/2014/09/funny-pictures-how-to-avoid-questions.jpg>
- PGP encryption image: https://static.goanywhere.com/images/products/mft/GoAnywhereMFT_OpenPGP-Diagram_web2018.png
- CVSS score image: <https://www.first.org/cvss/v3-1/media/dcbbdaef38f7d415ef9ccbd936d48d4e.png>
- JFK meme: <https://imgflip.com/i/3si67b>
- Private sign: <https://veryfunnypics.eu/a-private-sign-2/>