Unique ways to Hack into a Python Web Service

YOURS TRULY

- Senior Solutions Engineer at we45
- Full Stack Developer
- Developer of Open source Projects
- Trainer and Speaker

Outline

- Why web services aren't secure
- Unique vulnerabilities
- Demo !
- Security Pipeline

Why web services aren't secure

API (Application Programming Interface) vulnerabilities are becoming more widespread as time goes by. Figure 4 shows the number of API vulnerabilities between 2015-2018. New API vulnerabilities in 2018 (264) increased by 23% over 2017 (214), by 56% compared to 2016 (169), and by 154% compared to 2015 (104).

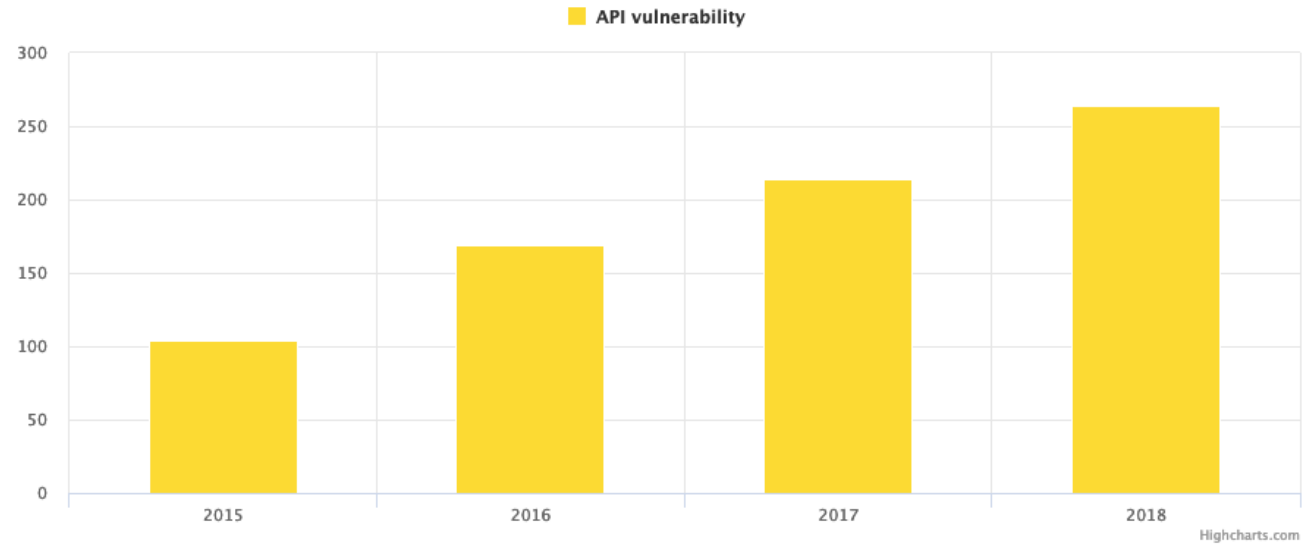
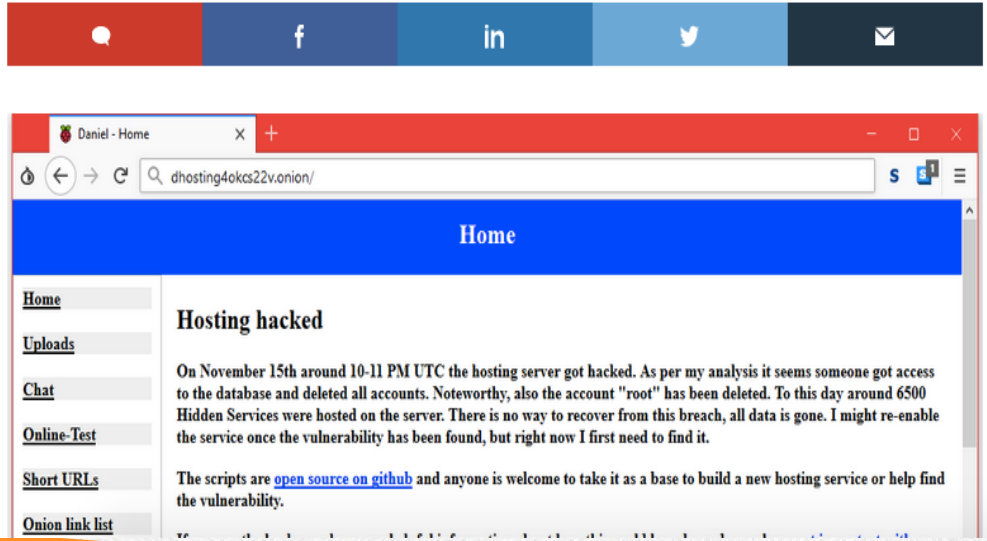


Figure 4: API vulnerabilities 2015-2018

Popular Dark Web hosting provider got hacked, 6,500 sites down

Hosting provider is still looking for the hacker's point of entry.

By Catalin Cimpanu for Zero Day | November 17, 2018 -- 21:39 GMT (03:09 IST) | Topic: Security



MORE FROM CATALIN CIMPANU

Security
Malware that hunts for account credentials on adult websites tripled in 2018

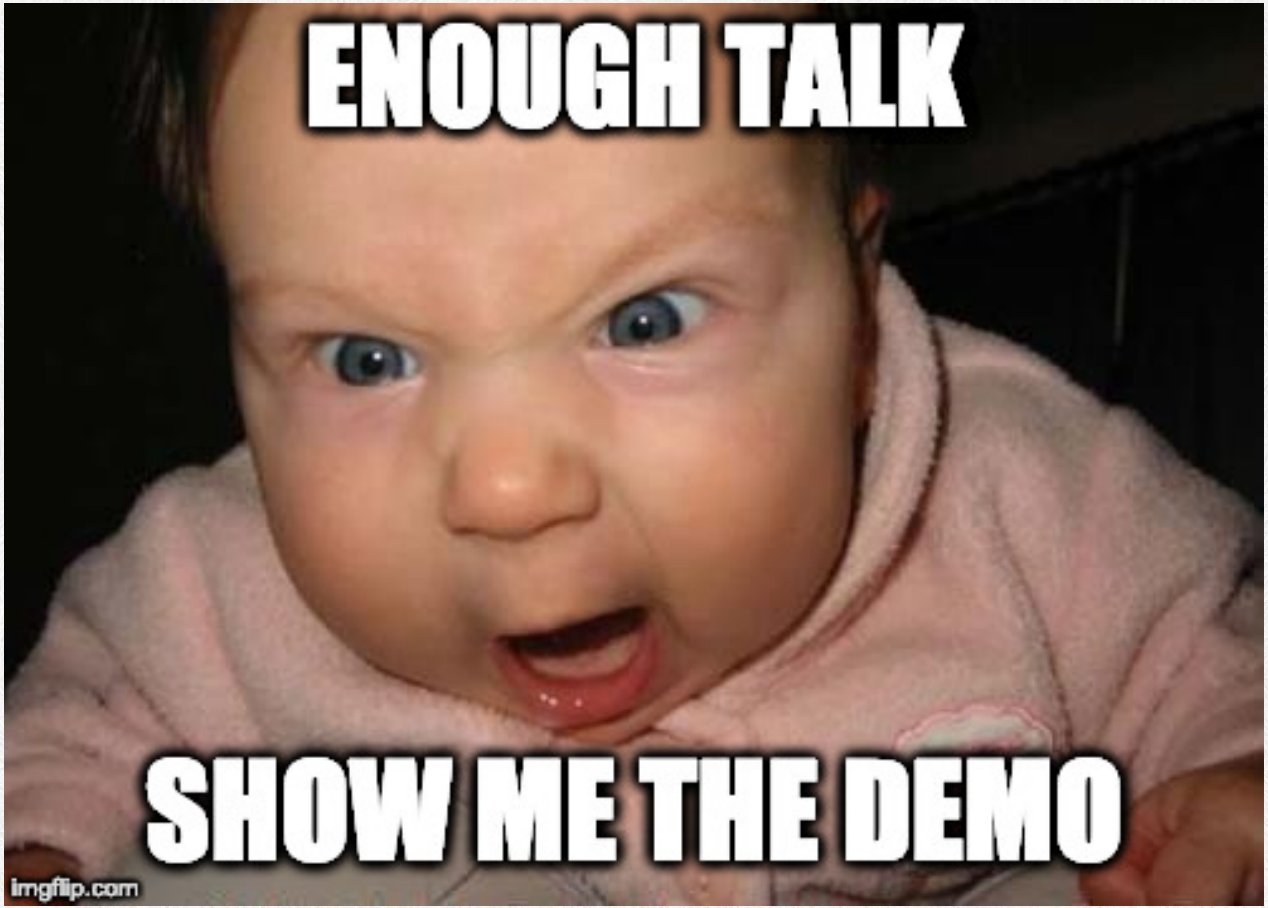
Security
A third of all Chrome extensions request access to user data on any site

Security
Microsoft publishes security alert on IIS bug that causes 100% CPU usage spikes

Security

Unique Vulnerabilities

- JWT Manipulation
- Insecure Deserialization
- Server Side Template Injection
- Click Jacking
- Etc …



JWT Manipulation

OWASP-2017 A5 Broken Access Control

Why JWT

- Stateless Application
- Authorization Mechanism
- Transfers information between server and client
- Scalable and decoupled

Encoded PASTE A TOKEN HERE

```
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjM0NTY3ODkwIiwibmFtZSI6IkpvaG4gRG9lIiwiaWF0IjoxNTE2MjM5MDIyfQ.SflKxwRJSMeKKF2QT4fwpMeJf36POk6yJV_adQssw5c
```

Decoded EDIT THE PAYLOAD AND SECRET

HEADER: ALGORITHM & TOKEN TYPE

```
{
  "alg": "HS256",
  "typ": "JWT"
}
```

PAYLOAD: DATA

```
{
  "sub": "1234567890",
  "name": "John Doe",
  "iat": 1516239022
}
```

VERIFY SIGNATURE

```
HMACSHA256(
  base64UrlEncode(header) + "." +
  base64UrlEncode(payload),
  your-256-bit-secret
```

Lots of ways to get JWT wrong

- Modify the algorithm to `none`
- Leakage of sensitive information
- Algorithm Confusion
- Cracking Secret Keys

CVE-2018-15801: Authorization Bypass During JWT Issuer Validation with spring-security

Severity

Low

Vendor

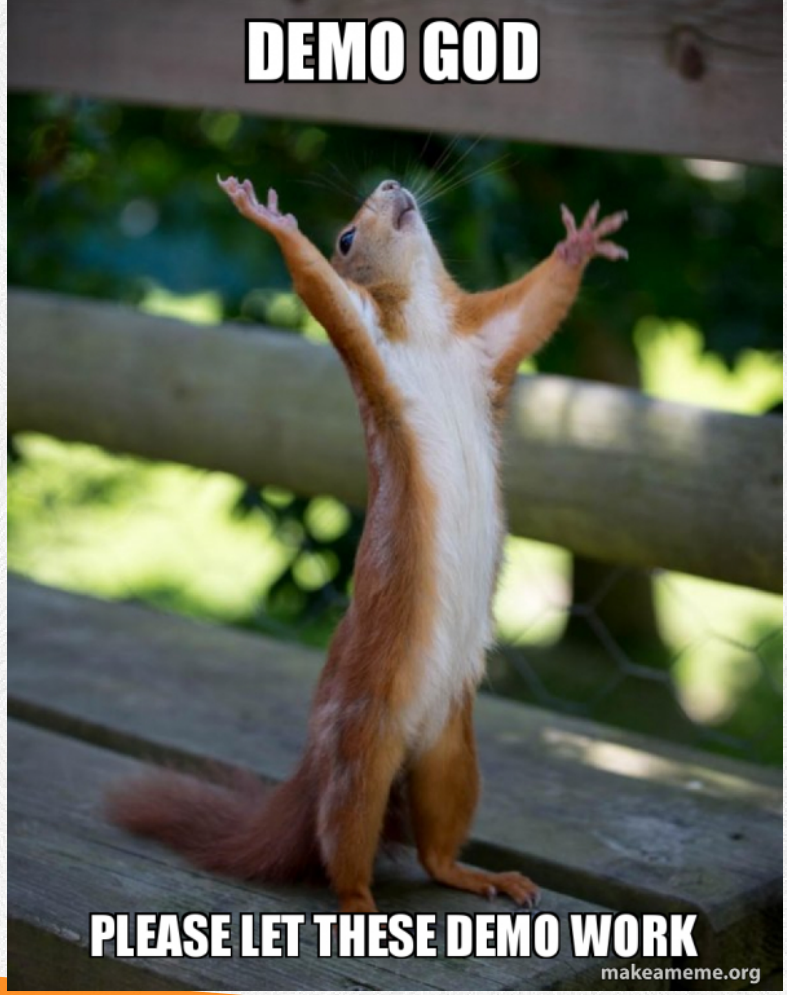
Spring by Pivotal

Description

Spring Security versions 5.1.x prior to 5.1.2 contain an authorization bypass vulnerability during JWT issuer validation. In order to be impacted, the same private key for an honest issuer and a malicious user must be used when signing JWTs. In that case, a malicious user could fashion signed JWTs with the malicious issuer URL that may be granted for the honest issuer.

Affected Pivotal Products and Versions

Severity is low unless otherwise noted.



Mitigation for JWT

- Know the Algorithms
- Ensure that JWT implementation doesn't support 'None' signature
- Secret key size must be strong
- JWT lifetime relatively short
- Check library flaw
- Validate using Unique ID

Server Side Template Injection

Why Template Engine

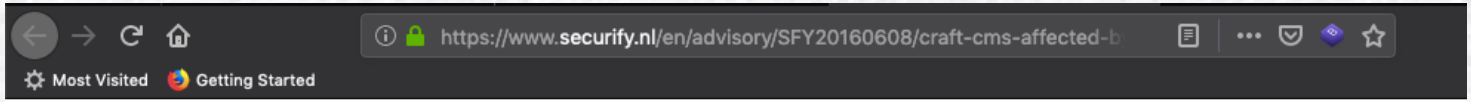
- Readability
- Separation
- Auto-Escaping
- Template Caching
- Template Inheritance

https://www.fifthdomain.com/civilian/2017/09/18/report-government-among-most-targeted-for-web-application-attacks/

> *regarded by users as highly trustworthy, the users of these sites — more than in other sectors — are unlikely to know the basics of how to stay safe online. This fact makes government sites tempting targets for Cross-Site Scripting attacks, which can infect a user's computer with malware. Another common type of attack in Q2 is Information Leak, which exploits various web application vulnerabilities in order to obtain additional data about users, the system itself, and other sensitive information.*

Within the critical infrastructure sectors of energy and manufacturing, the data diverged slightly from cross-industry averages. SQL injection (48.1 percent), operating system commanding (36.4 percent), server-side template injection (7.8 percent) and path traversal (5.4 percent) were most common. The different methods can be explained by attackers' motives:

> *By contrast, in the case of energy and manufacturing companies, attackers' objective is to obtain full control over company infrastructure. Therefore the most common attacks attempt to run arbitrary OS commands and gain control over the server or obtain information about the system; attacks on users are few and far between. By launching attacks against the target company's internal network, an attacker can gain access to critical system components and interfere with operations.*



Craft CMS affected by server side template injection

Abstract

It was discovered that Craft CMS is vulnerable to server-side template injection. An authenticated attacker can exploit this issue to compromise Craft CMS, for example by retrieving sensitive data from configuration files.

Tested versions

All versions of Craft CMS prior to build 2791 are affected by this vulnerability.

Fix

Pixel & Tonic, Inc. released Craft CMS build 2791 that resolves this vulnerability. This build can easily be installed through the Control Panel. After the fix is applied the rendering of templates is globally limited in *TemplatesService.php* and *TwigEnvironment.php*.

Introduction

Craft CMS is a Content Management System comparable to WordPress or Drupal. Craft CMS is written in PHP and uses the Yii and Twig frameworks. When users update their profile, they are redirected to (parts of) the Control Panel.

 @ti1akt

Mitigation

- Sanitise and then render
- Sandboxing the application

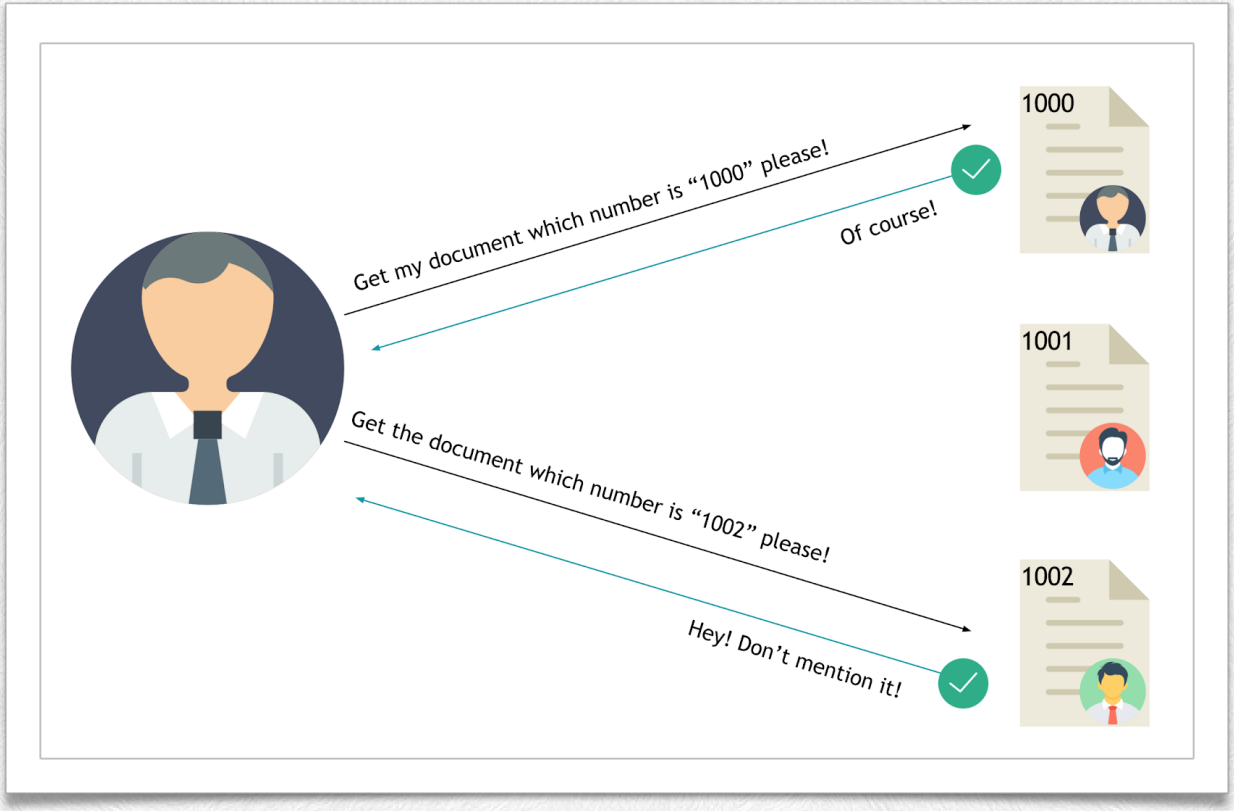
Insecure Direct Object Reference

OWASP-2017 A8 Broken Access Control

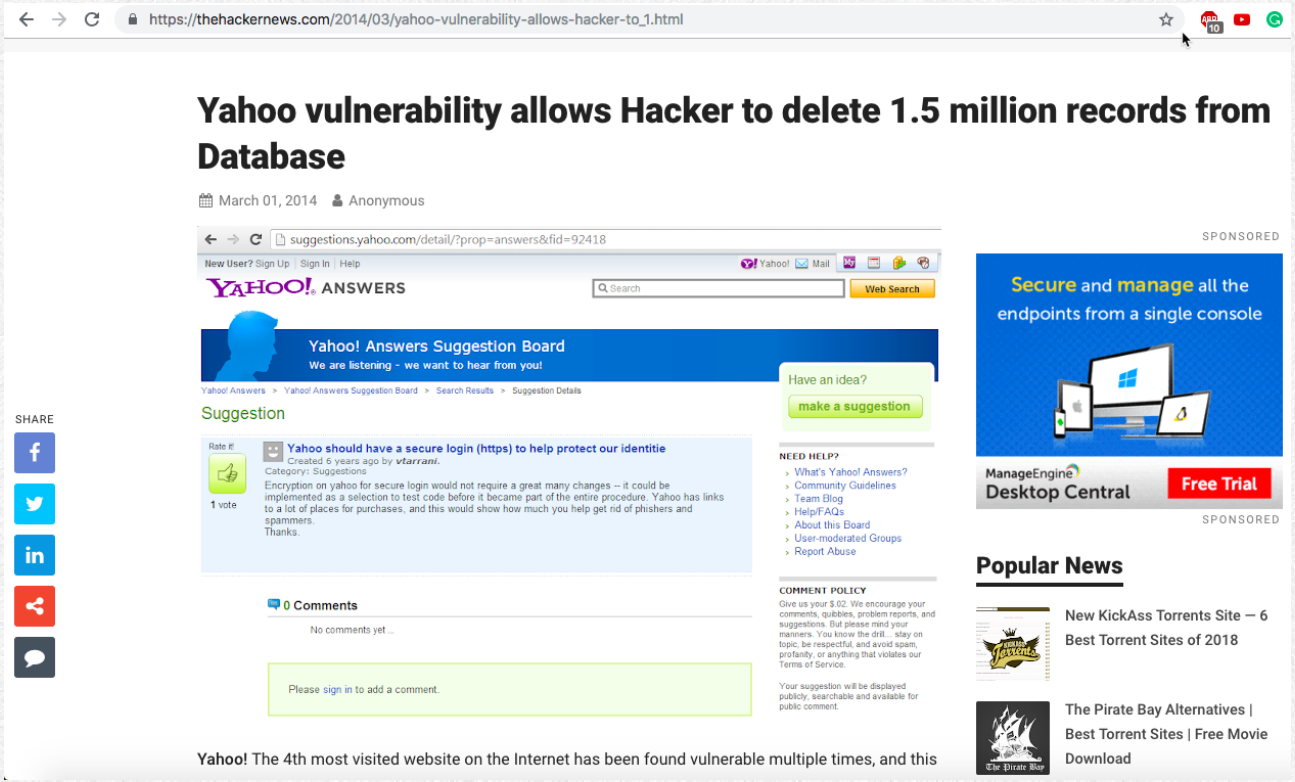
Insecure Direct Object reference

- “id”,”pid”,”uid” are often seen in HTTP parameter
- Accessing other user privilege
- Backend not properly validated users

How it works



Yahoo Breach



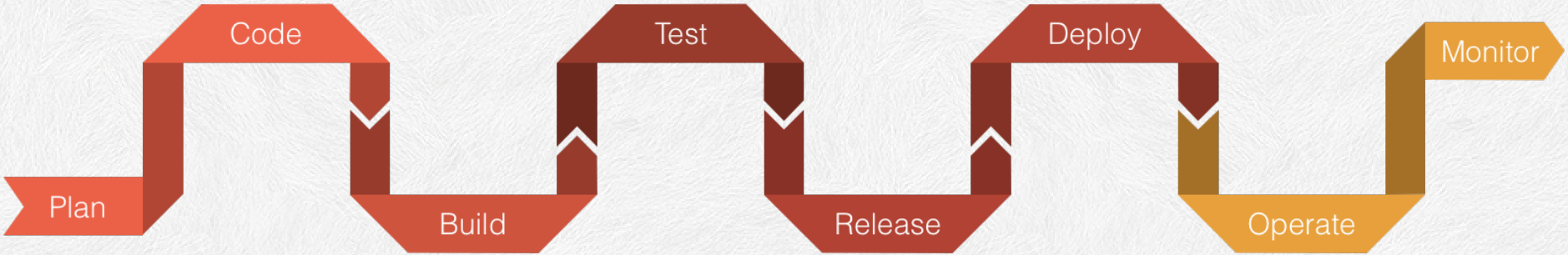
 @ti1akt

Source:https://thehackernews.com/2014/03/yahoo-vulnerability-allows-hacker-to_1.html

Mitigation

- Validate user using requested query
- Check database is that user is genuine or not
- Custom validation in server side as well as client side
- JWT should be invalidated once the user is logout

 @ti1akt



SAST Security - Composition	DAST IAST	Security in IaC	Security monitoring & attack detection

Basic Pipeline Demo

SecDevOps Jenkins pipeline

@ti1akt

Thank you


@ti1akt

https://github.com/we45/pyconth2019


ti1akt