OWASP
Open Web Application
Security Project

Automating API Pen Testing using Fuzzapi

just another tool?

About us

Abhijeth Dugginapeddi
@abhijeth
Application Security
Likes training, spreading awareness
Got some bugs in Google/FB/Yahoo/Microsoft etc
Among top 5 bug hunters on Synack

Srinivas Rao Kotipalli
@srini0x00
Security Engineer
Author, Speaker, Trainer
Blogs at androidpentesting.com
Author of “Hacking Android”

Lalith Rallabhandi
@lalithr95
Developer Intern
Blogger, Coder, Security Enthusiast
Does bounties when free and found bugs
With Microsoft/Google/FB/Badoo etc

Only @abhijeth @srini0x00 and @lalithr95 are responsible for whatever is on the slides

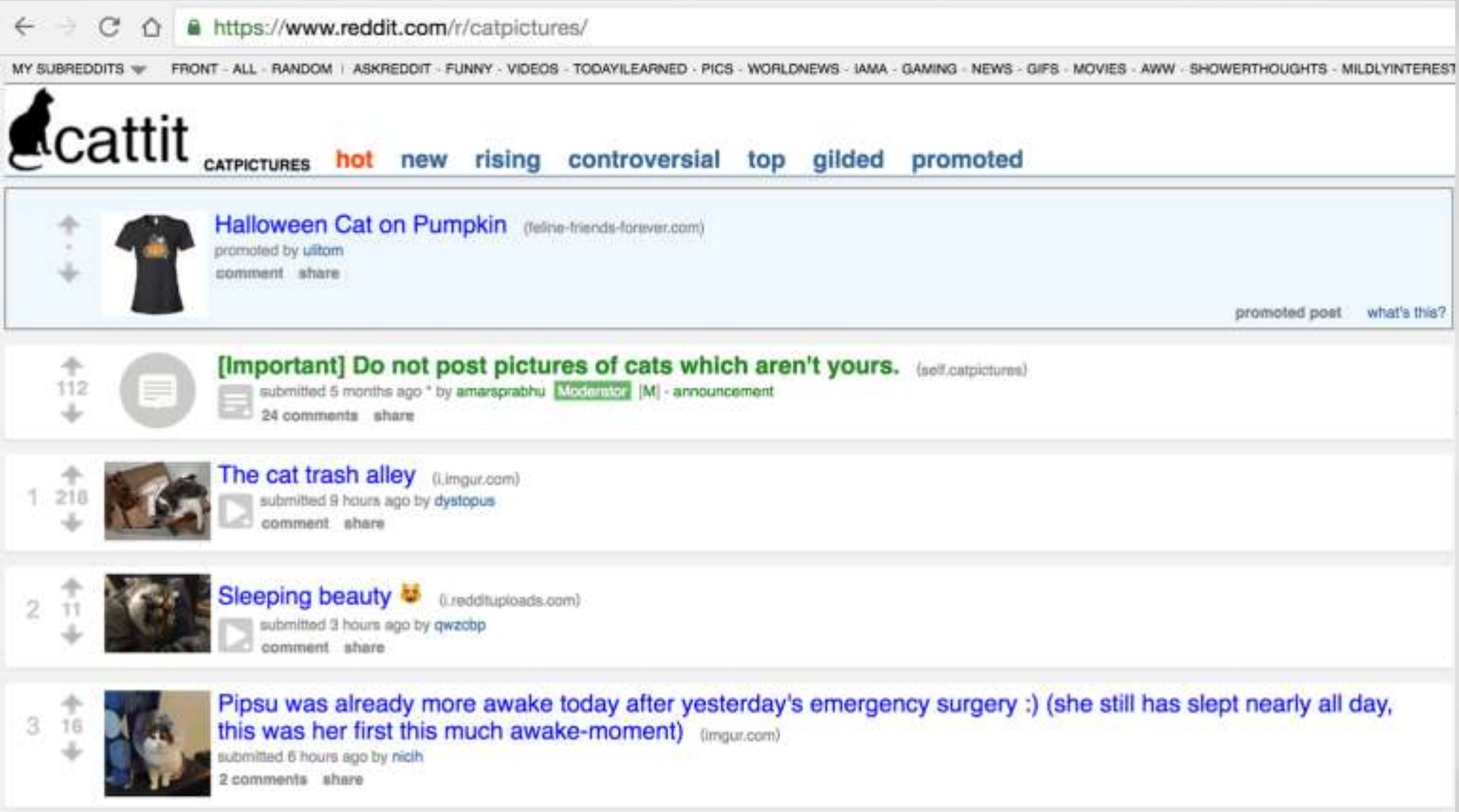
Nobody else is responsible for anything else we say

Next 45 minutes

-Why
-What
-How

Source giphy

Source
http://vignette2.wikia.nocookie.net/garfield/images/4/43/Garfield_the_Cat.png/revision/latest?cb=20150508141623


https://www.reddit.com/r/catpictures/
MY SUBREDDITS
FRONT - ALL - RANDOM | ASKREDDIT - FUNNY - VIDEOS - TODAYILEARNED - PICS - WORLDNEWS - IAMA - GAMING - NEWS - GIFS - MOVIES - AWW - SHOWERTHOUGHTS - MILDLYINTEREST
cattit
CATPICTURES
hot
new
rising
controversial
top
gilded
promoted

Halloween Cat on Pumpkin (feline-friends-forever.com)
promoted by ulltom
comment share
promoted post what's this?

112 [Important] Do not post pictures of cats which aren't yours. (self.catpictures)
submitted 5 months ago * by amarsprabhu Moderator [M] - announcement
24 comments share

1 218 The cat trash alley (i.imgur.com)
submitted 9 hours ago by dystopus
comment share

2 11 Sleeping beauty 🐱 (i.reddituploads.com)
submitted 3 hours ago by qwzobp
comment share

3 16 Pipsu was already more awake today after yesterday's emergency surgery :) (she still has slept nearly all day, this was her first this much awake-moment) (imgur.com)
submitted 6 hours ago by nicih
2 comments share

Source reddit

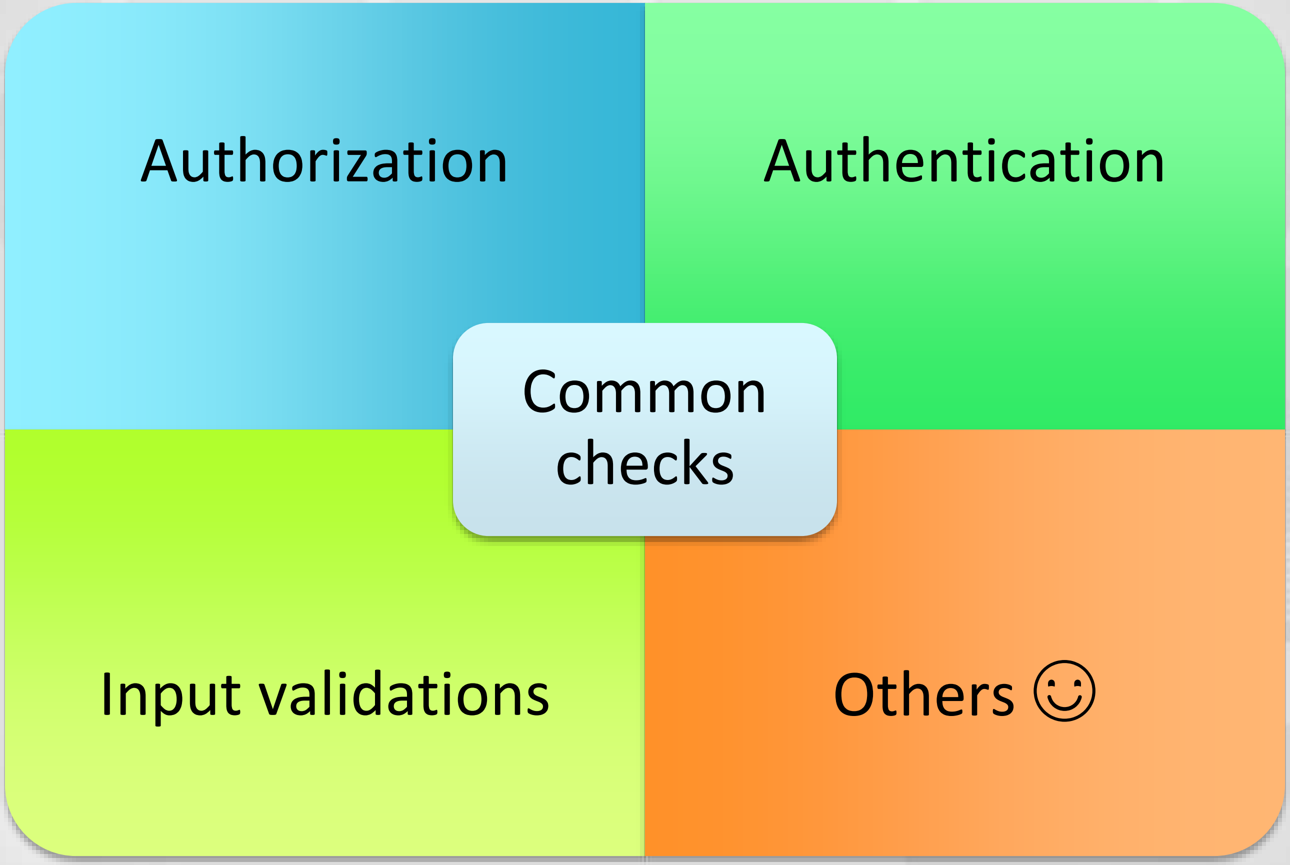
On a serious note

- What is fuzzAPI
- How to use fuzzAPI
- Need for automating Pen Testing APIs
- Developer vs Pen tester use cases
- Continuous Integration
- Spread the smile ☺

#fuzzAPI

- Open Source REST API Fuzzer
- Test for vulnerabilities while writing your code
- Helps Pen testers to fasten their testing
- Covers most top attacks on APIs
- Built in Ruby on Rails

Rest API Penetration Testing



Authorization

Authentication

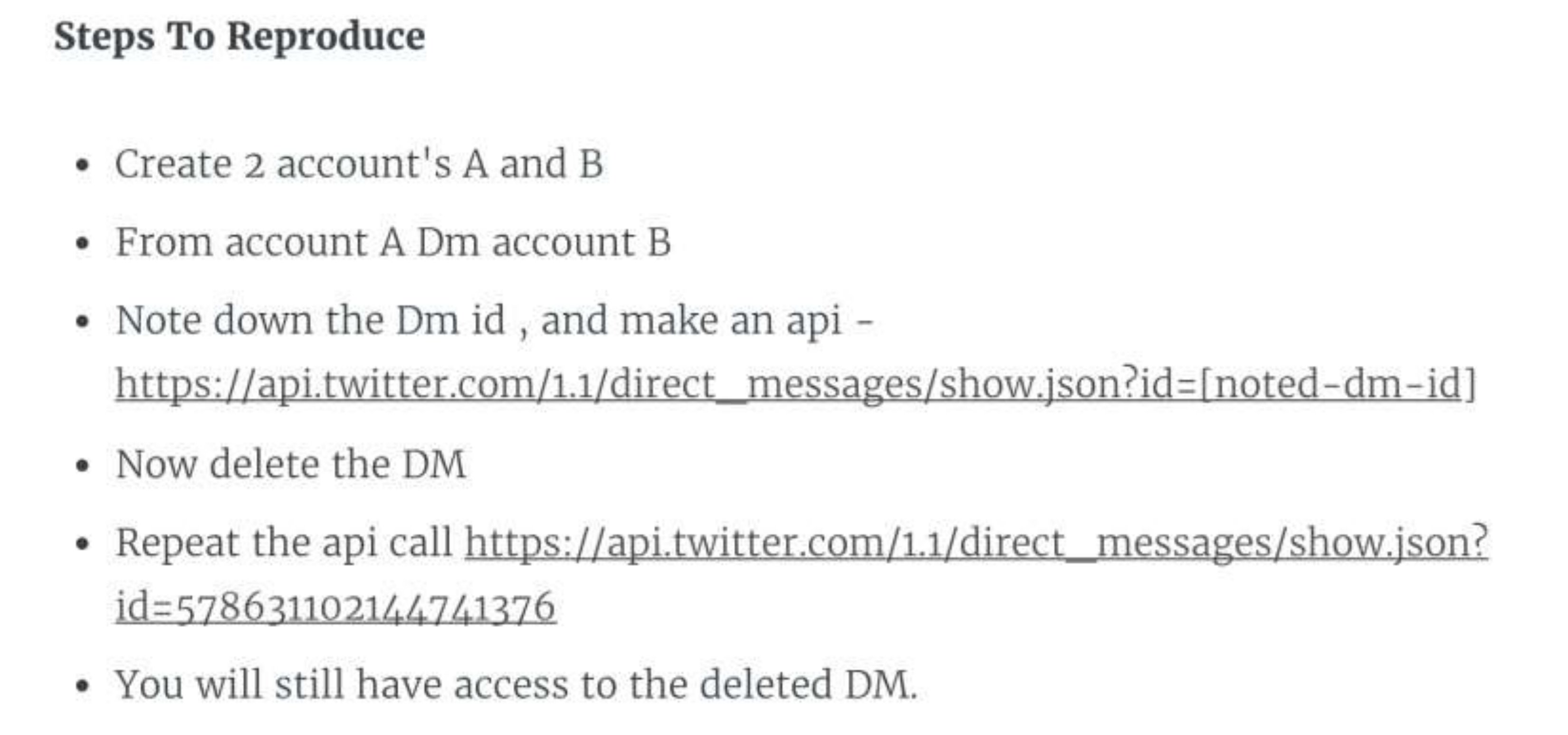
Common checks

Input validations

Others ☺

#welovebugs

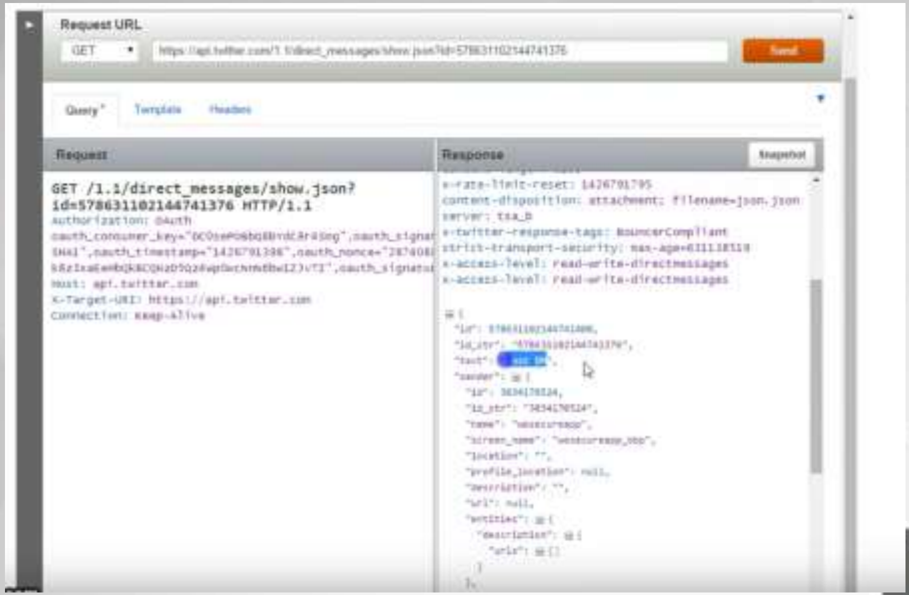
This is Twitter



Steps To Reproduce

- Create 2 account's A and B
- From account A Dm account B
- Note down the Dm id , and make an api - https://api.twitter.com/1.1/direct_messages/show.json?id=[noted-dm-id]
- Now delete the DM
- Repeat the api call https://api.twitter.com/1.1/direct_messages/show.json?id=578631102144741376
- You will still have access to the deleted DM.

Source: @wesecureapp

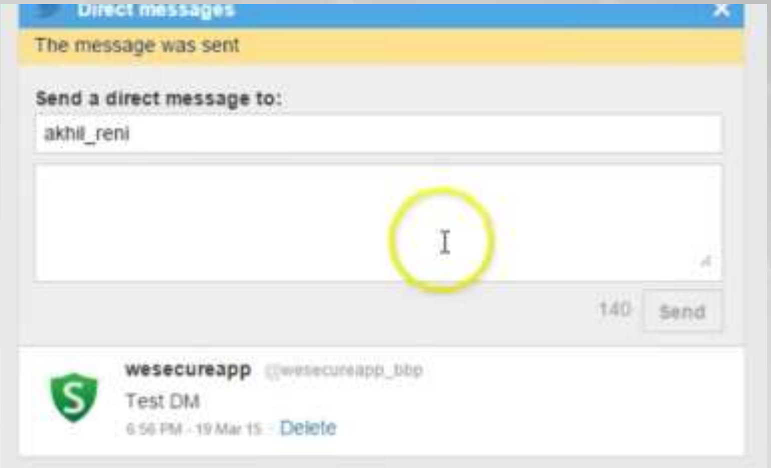


https://api.twitter.com/1.1/direct_messages/show.json?id=578631102144741376

```
GET /1.1/direct_messages/show.json?
id=578631102144741376 HTTP/1.1
Authorization: OAuth
oauth_consumer_key="DC0sePOBbQ8bYdC8r4Smg",oauth_signat
SHA1",oauth_timestamp="1426791398",oauth_nonce="287408
k8z1xaEeHbQkBCQN2D5Q24wpOwCNnN6bw1ZJvTI",oauth_signatu
Host: api.twitter.com
X-Target-URI: https://api.twitter.com
Connection: Keep-Alive
```

```
x-rate-limit-reset: 1426791795
content-disposition: attachment; filename=json.json
server: tsa_b
x-twitter-response-tags: BouncerCompliant
strict-transport-security: max-age=631138519
x-access-level: read-write-directmessages
x-access-level: read-write-directmessages
```

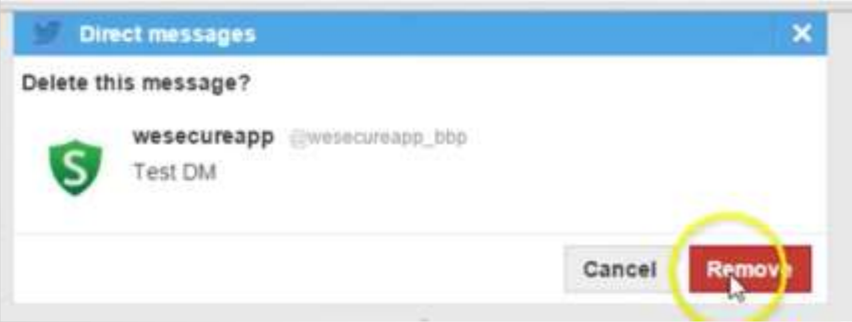
```
{
  "id": 578631102144741400,
  "id_str": "578631102144741376",
  "text": "Test DM",
  "sender": {
    "id": 3034170524,
    "id_str": "3034170524",
    "name": "wesecureapp",
    "screen_name": "wesecureapp_bbp",
    "location": "",
    "profile_location": null,
    "description": "",
    "url": null,
    "entities": {
      "description": {
        "urls": []
      }
    },
```



The message was sent

Send a direct message to: akhil_reni

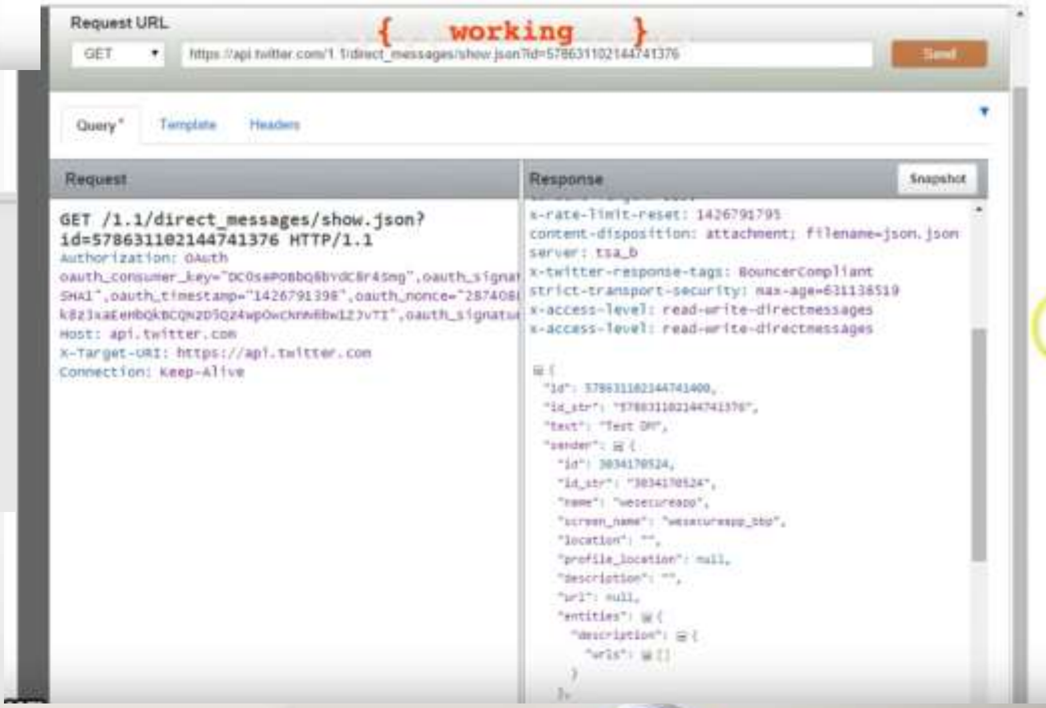
wesecureapp @wesecureapp_bbp
Test DM
6:56 PM - 19 Mar 15



Delete this message?

wesecureapp @wesecureapp_bbp
Test DM

Source: @wesecureapp



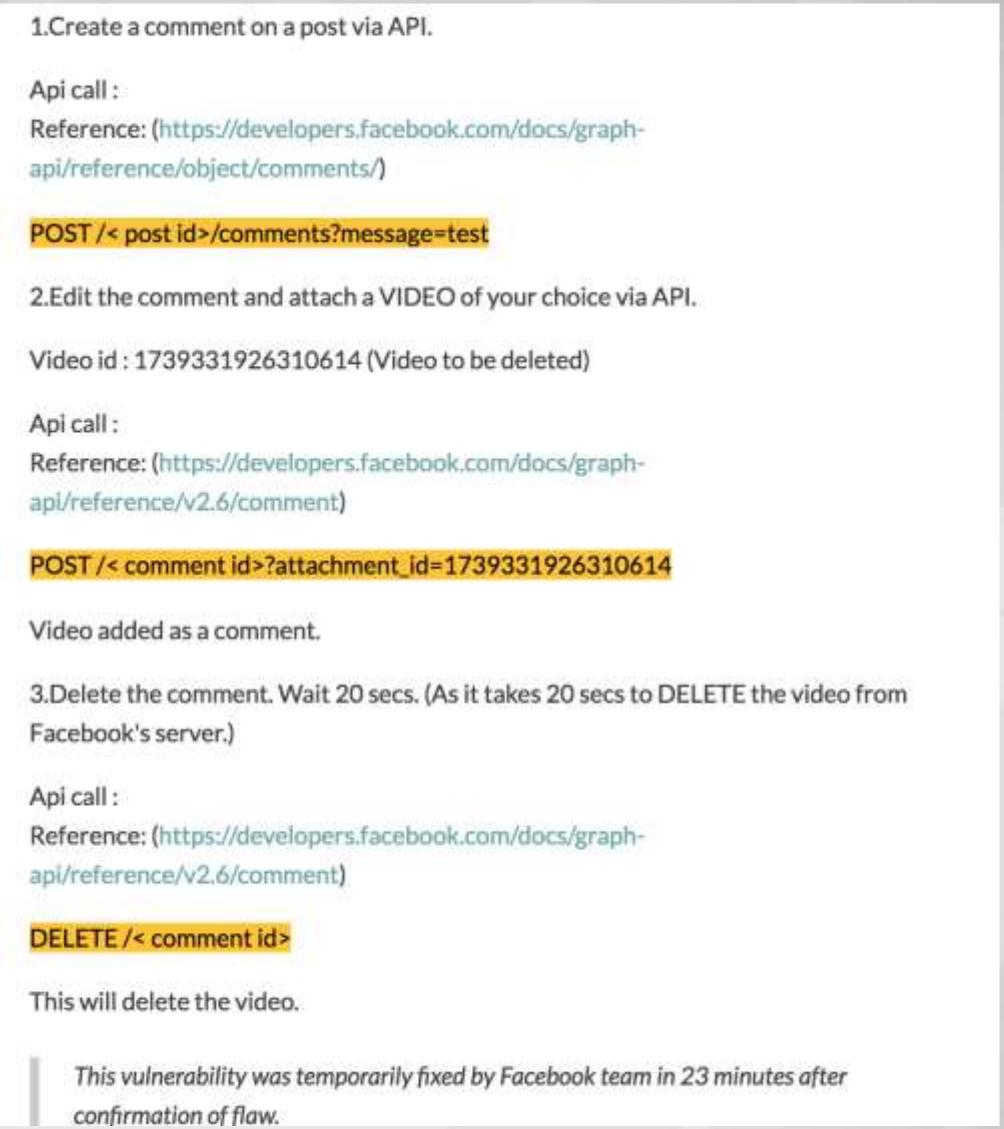
https://api.twitter.com/1.1/direct_messages/show.json?id=578631102144741376

```
GET /1.1/direct_messages/show.json?
id=578631102144741376 HTTP/1.1
Authorization: OAuth
oauth_consumer_key="DC0sePOBbQ8bYdC8r4Smg",oauth_signat
SHA1",oauth_timestamp="1426791398",oauth_nonce="287408
k8z1xaEeHbQkBCQN2D5Q24wpOwCNnN6bw1ZJvTI",oauth_signatu
Host: api.twitter.com
X-Target-URI: https://api.twitter.com
Connection: Keep-Alive
```

```
x-rate-limit-reset: 1426791795
content-disposition: attachment; filename=json.json
server: tsa_b
x-twitter-response-tags: BouncerCompliant
strict-transport-security: max-age=631138519
x-access-level: read-write-directmessages
x-access-level: read-write-directmessages
```

```
{
  "id": 578631102144741400,
  "id_str": "578631102144741376",
  "text": "Test DM",
  "sender": {
    "id": 3034170524,
    "id_str": "3034170524",
    "name": "wesecureapp",
    "screen_name": "wesecureapp_bbp",
    "location": "",
    "profile_location": null,
    "description": "",
    "url": null,
    "entities": {
      "description": {
        "urls": []
      }
    },
```

Facebook ☺



1.Create a comment on a post via API.

Api call :
Reference: (https://developers.facebook.com/docs/graph-api/reference/object/comments/)

POST /< post id>/comments?message=test

2.Edit the comment and attach a VIDEO of your choice via API.

Video id : 1739331926310614 (Video to be deleted)

Api call :
Reference: (https://developers.facebook.com/docs/graph-api/reference/v2.6/comment)

POST /< comment id>?attachment_id=1739331926310614

Video added as a comment.

3.Delete the comment. Wait 20 secs. (As it takes 20 secs to DELETE the video from Facebook's server.)

Api call :
Reference: (https://developers.facebook.com/docs/graph-api/reference/v2.6/comment)

DELETE /< comment id>

This will delete the video.

> *This vulnerability was temporarily fixed by Facebook team in 23 minutes after confirmation of flaw.*

Credits: www.**pranavhivarekar**.in

Interesting?

Can you automate such attacks?

CONNECT LEARN GROW

May be!!

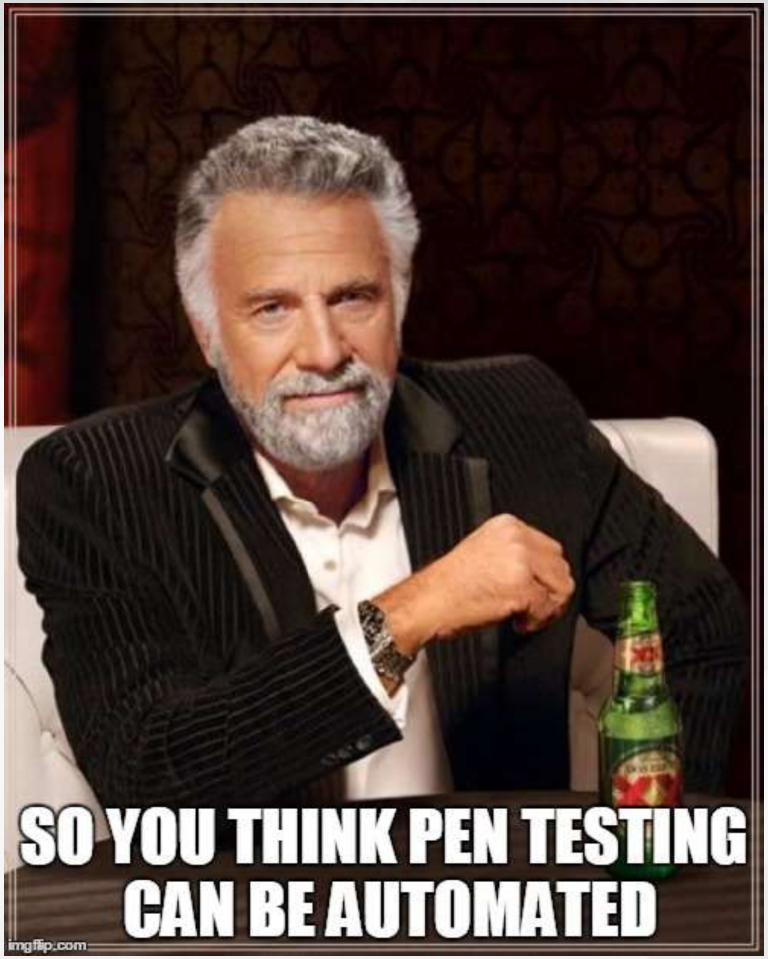
But why do you want to automate?

People don’t have time

Source: giphy

Continuous Integration

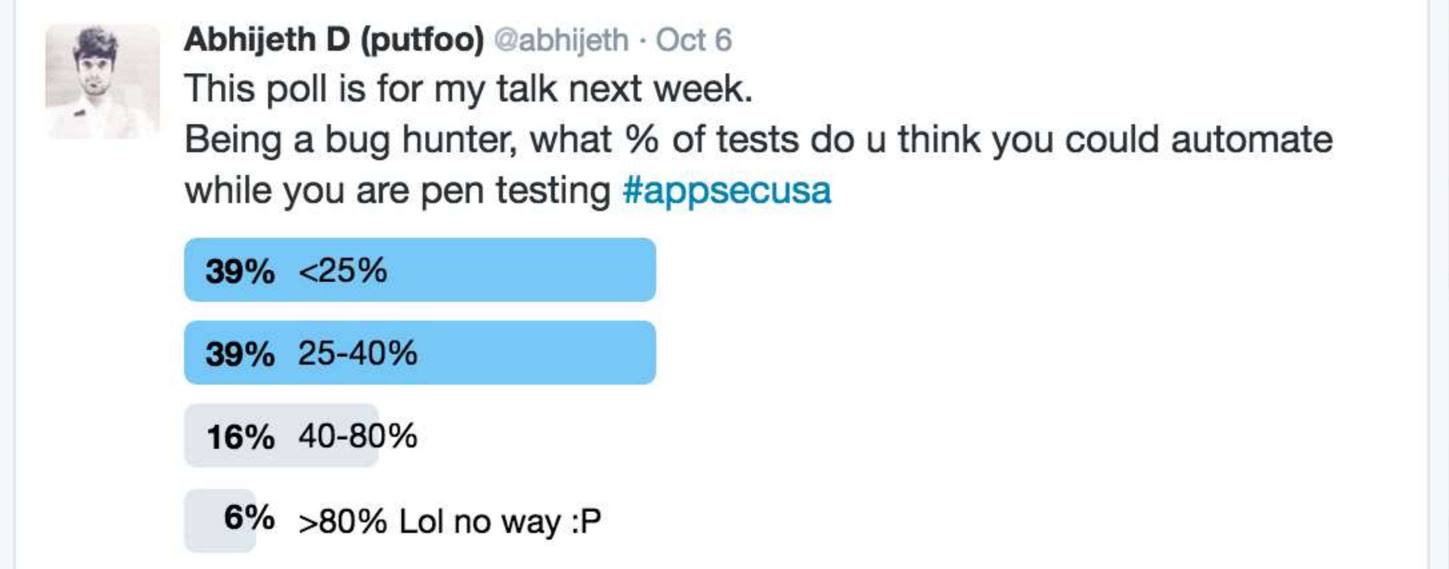
- There are companies/teams who deploy code to production >10 times every day
- Developers can do basic testing
- Penetration testers can save a lot of time
- Penetration testers can work on logical stuff
- Easier to fix vulnerabilities sooner than later



Source memegenerator

No

But a part of it can be automated.



Abhijeth D (putfoo) @abhijeth · Oct 6

This poll is for my talk next week.
Being a bug hunter, what % of tests do u think you could automate while you are pen testing #appsecusa

39% <25%

39% 25-40%

16% 40-80%

6% >80% Lol no way :P

Cool stuff about Fuzzapi

Access Control Violation

Privilege Escalation

XXE

Rate limiting

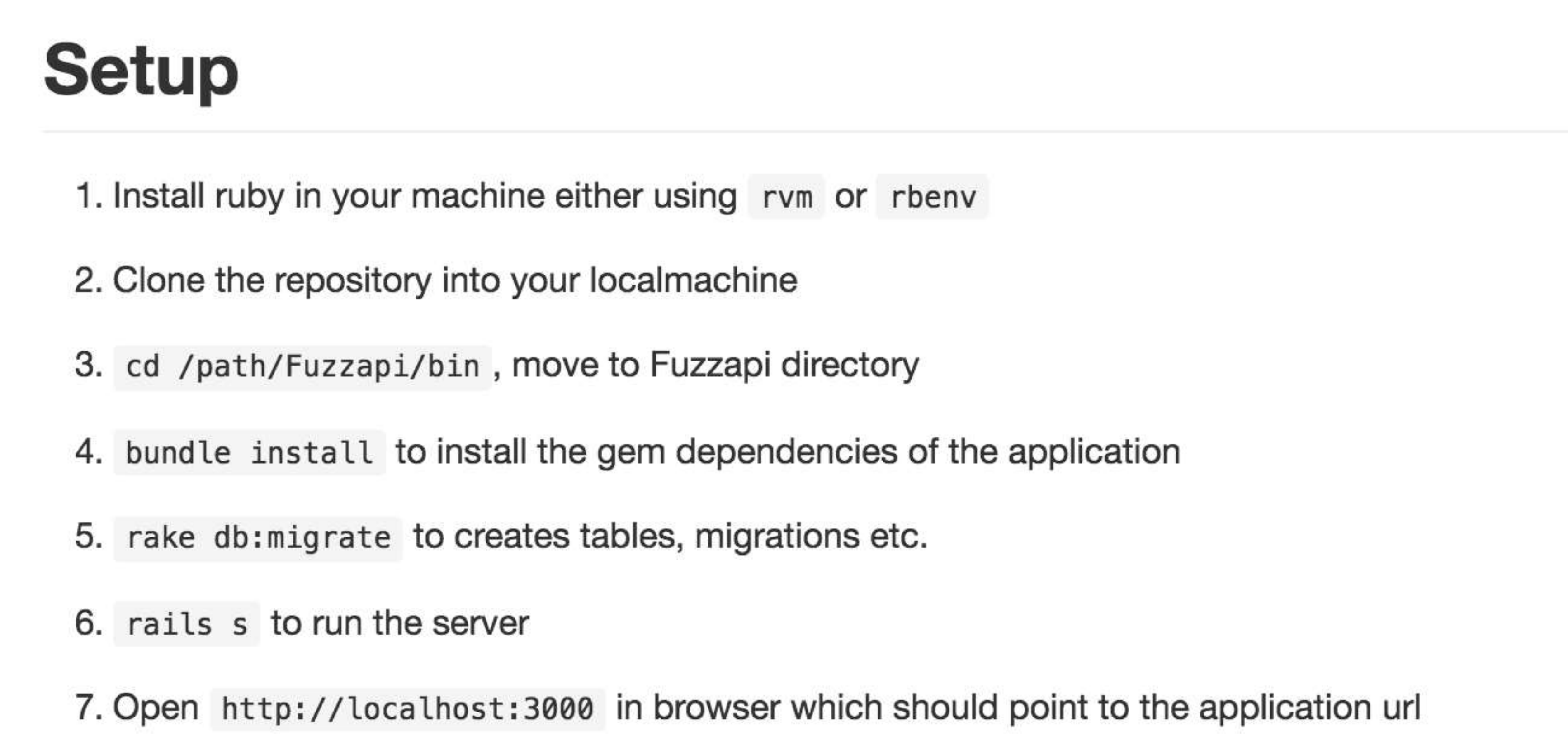
Other regular vulns like XSS/SQLi.. etc

Not so cool stuff!!

Demo

Source memegenerator

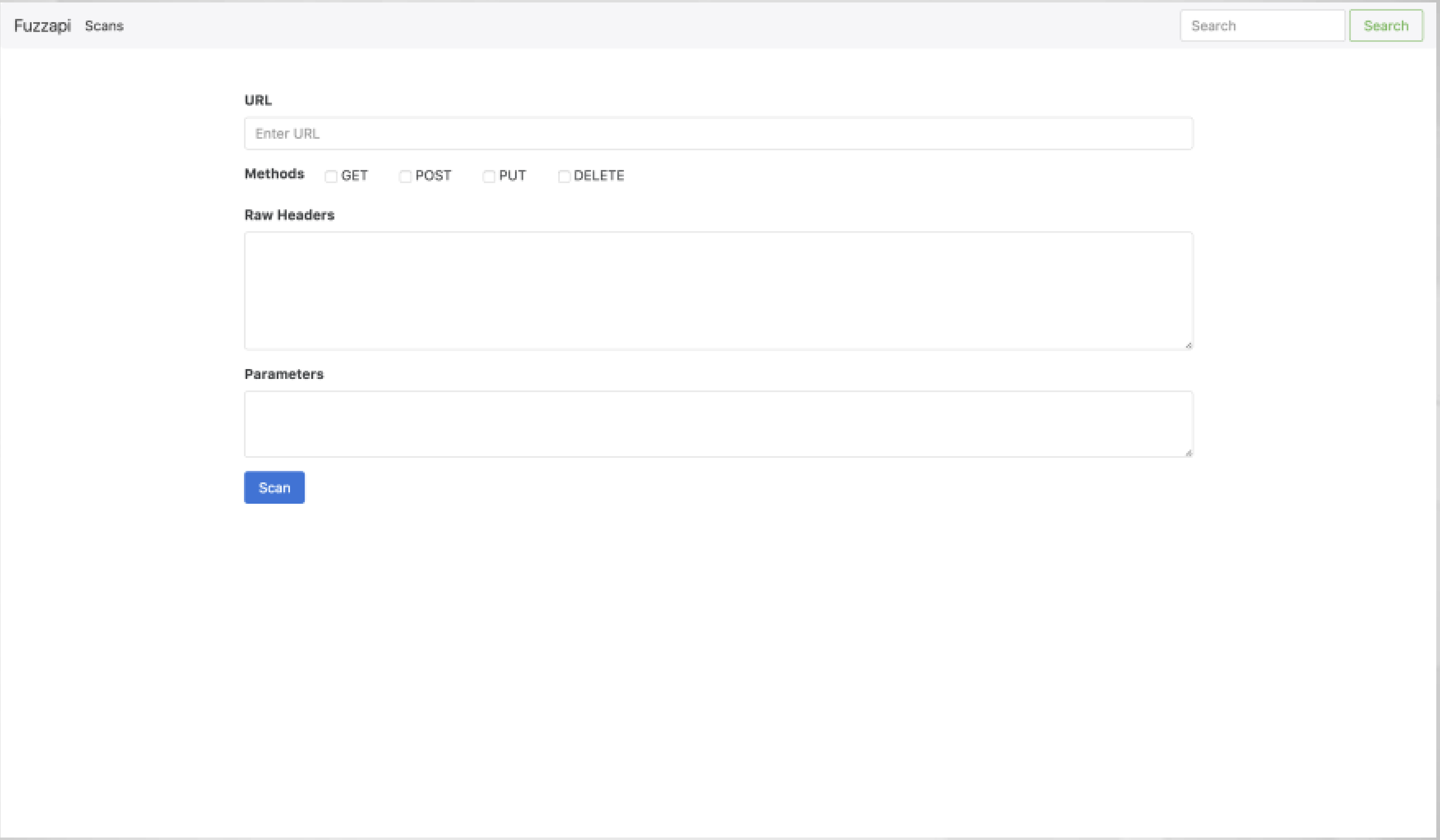
#if demo doesn’t work



Setup

1. Install ruby in your machine either using `rvm` or `rbenv`
2. Clone the repository into your localmachine
3. `cd /path/Fuzzapi/bin` , move to Fuzzapi directory
4. `bundle install` to install the gem dependencies of the application
5. `rake db:migrate` to creates tables, migrations etc.
6. `rails s` to run the server
7. Open `http://localhost:3000` in browser which should point to the application url

#if demo doesn't work



URL

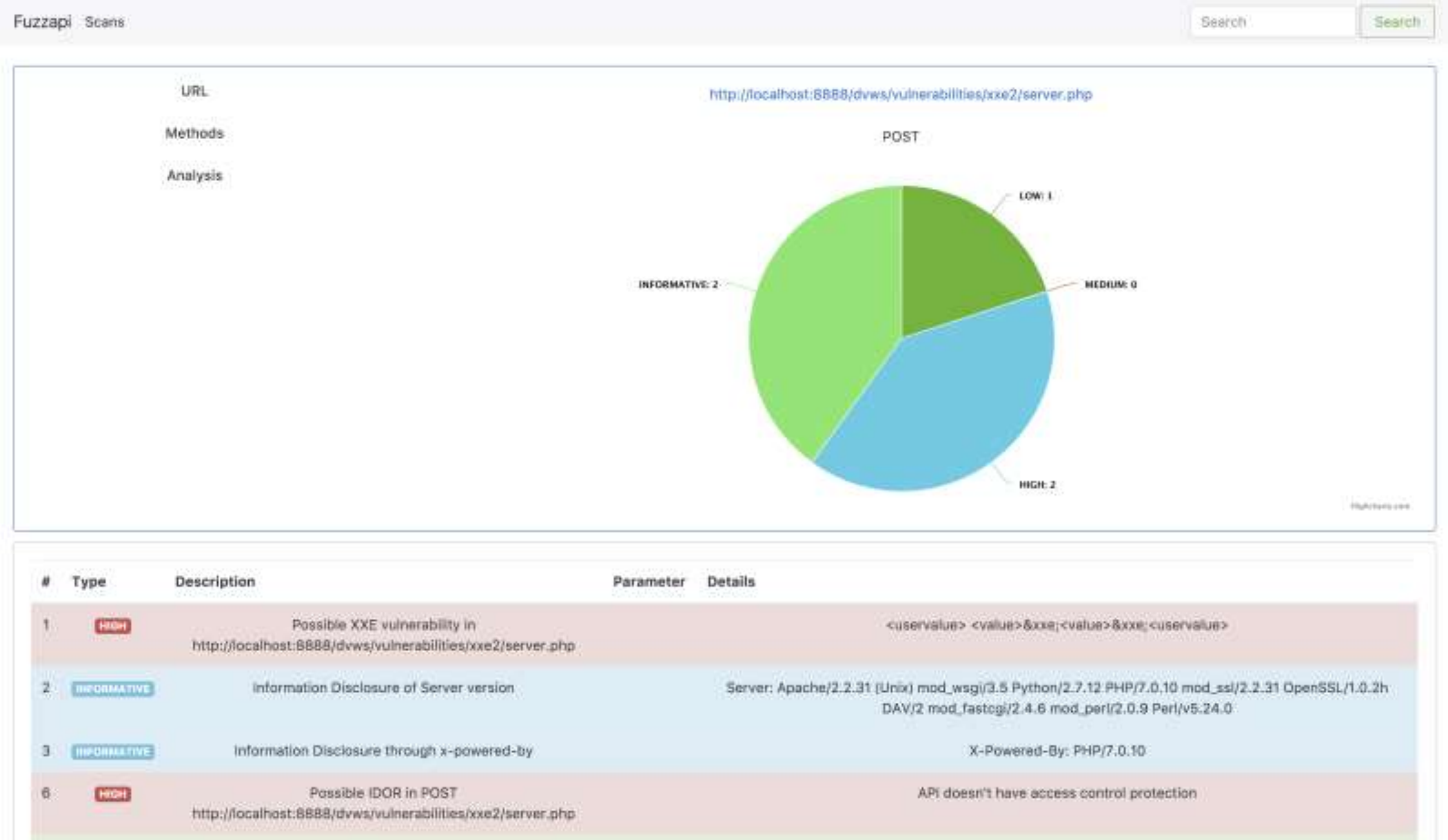
Enter URL

Methods □ GET □ POST □ PUT □ DELETE

Raw Headers

Parameters

#if demo doesn't work



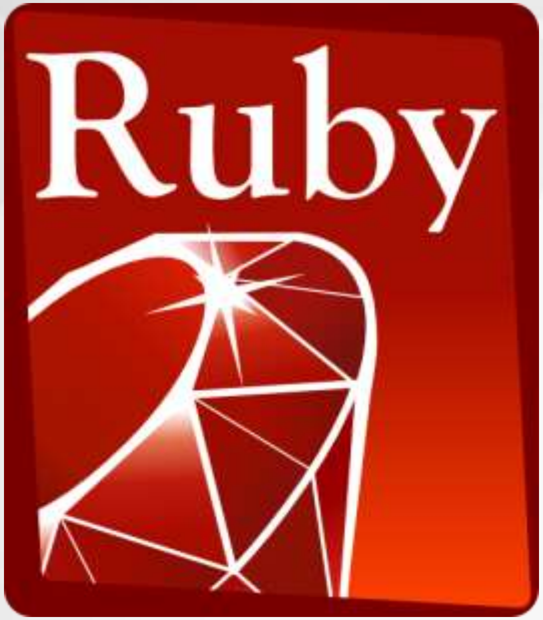
URL: http://localhost:8888/dvws/vulnerabilities/xxe2/server.php

Methods: POST

Analysis

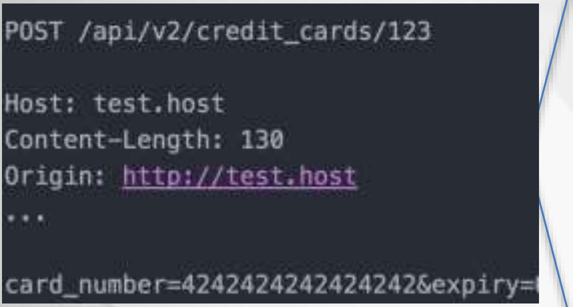
#	Type	Description	Parameter	Details
1	HIGH	Possible XXE vulnerability in http://localhost:8888/dvws/vulnerabilities/xxe2/server.php		<uservalue> <value>&xxe;<value>&xxe;<uservalue>
2	INFORMATIVE	Information Disclosure of Server version		Server: Apache/2.2.31 (Unix) mod_wsgi/3.5 Python/2.7.12 PHP/7.0.10 mod_ssl/2.2.31 OpenSSL/1.0.2h DAV/2 mod_fastcgi/2.4.6 mod_perl/2.0.9 Perl/v5.24.0
3	INFORMATIVE	Information Disclosure through x-powered-by		X-Powered-By: PHP/7.0.10
6	HIGH	Possible IDOR in POST http://localhost:8888/dvws/vulnerabilities/xxe2/server.php		API doesn't have access control protection

How stuff works

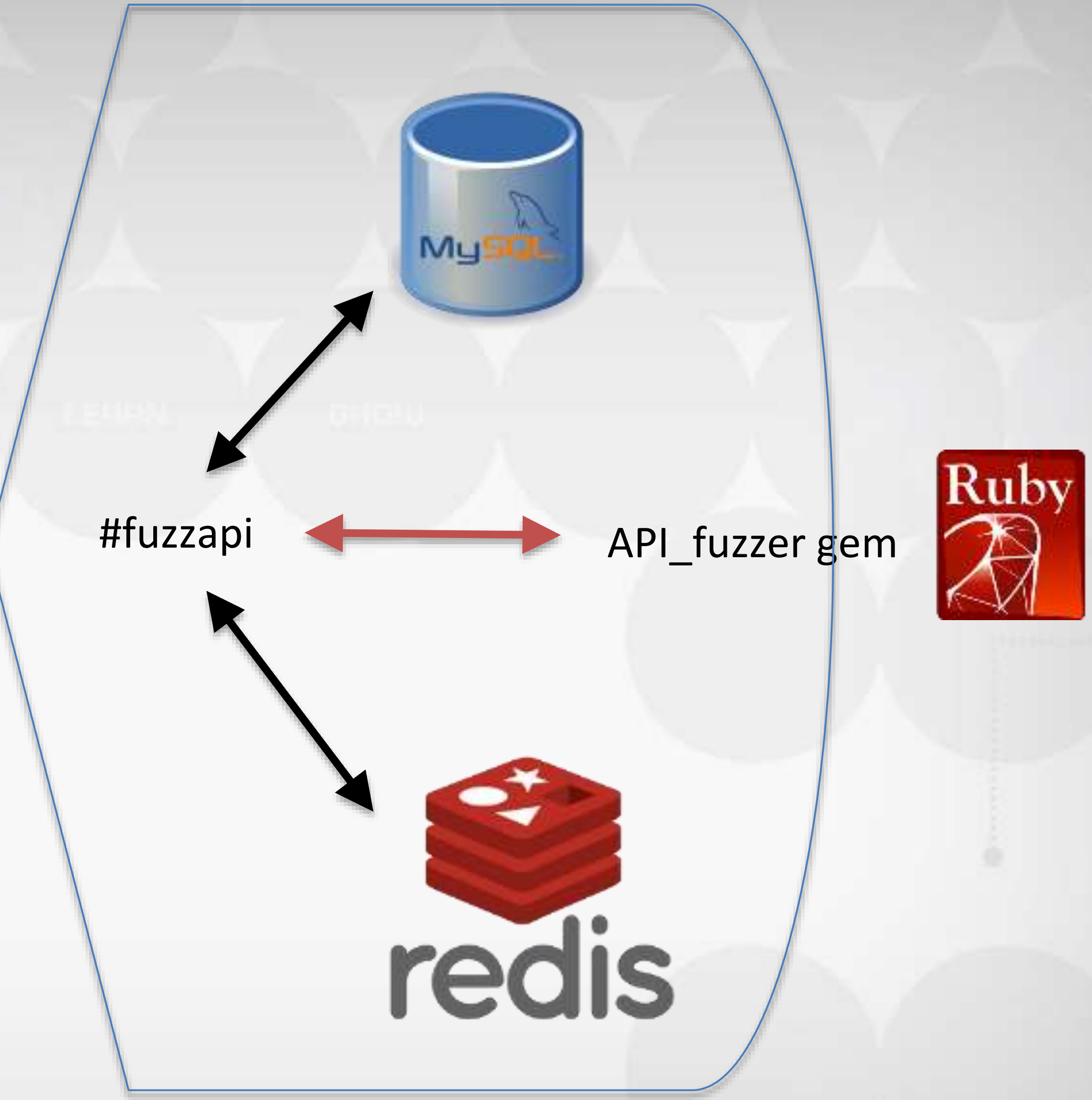


API_Fuzzer – Ruby gem

Fuzzapi -- Rails application



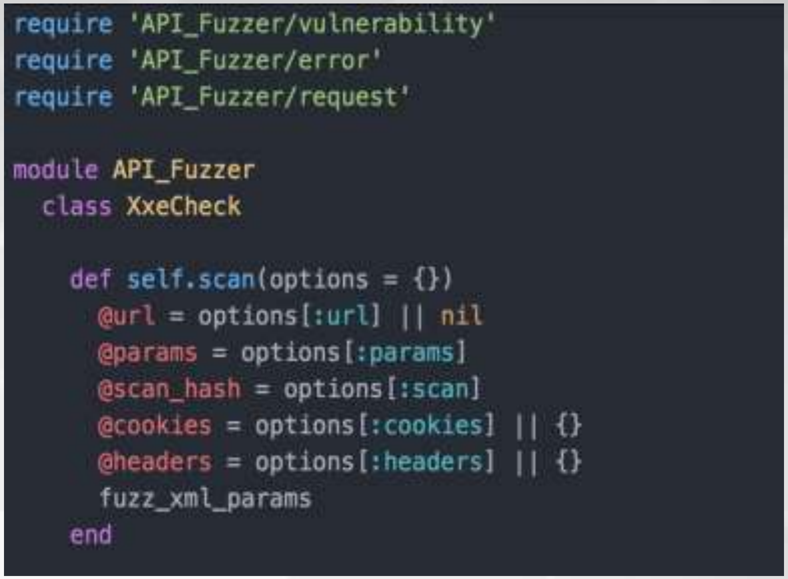
```
POST /api/v2/credit_cards/123

Host: test.host
Content-Length: 130
Origin: http://test.host
...

card_number=4242424242424242&expiry=
```



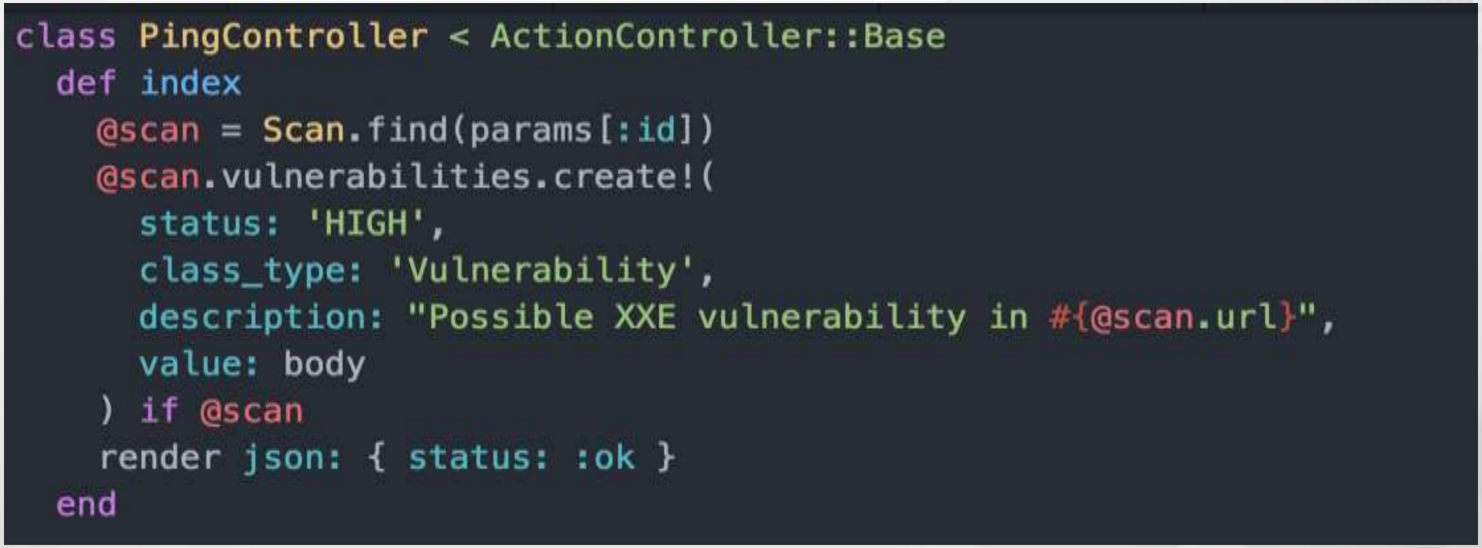
Code walk through

Fuzzapi approach for XXE

- XxeCheck performs a call with payload to internal server
- If status: OK – fuzzapi confirms XXE

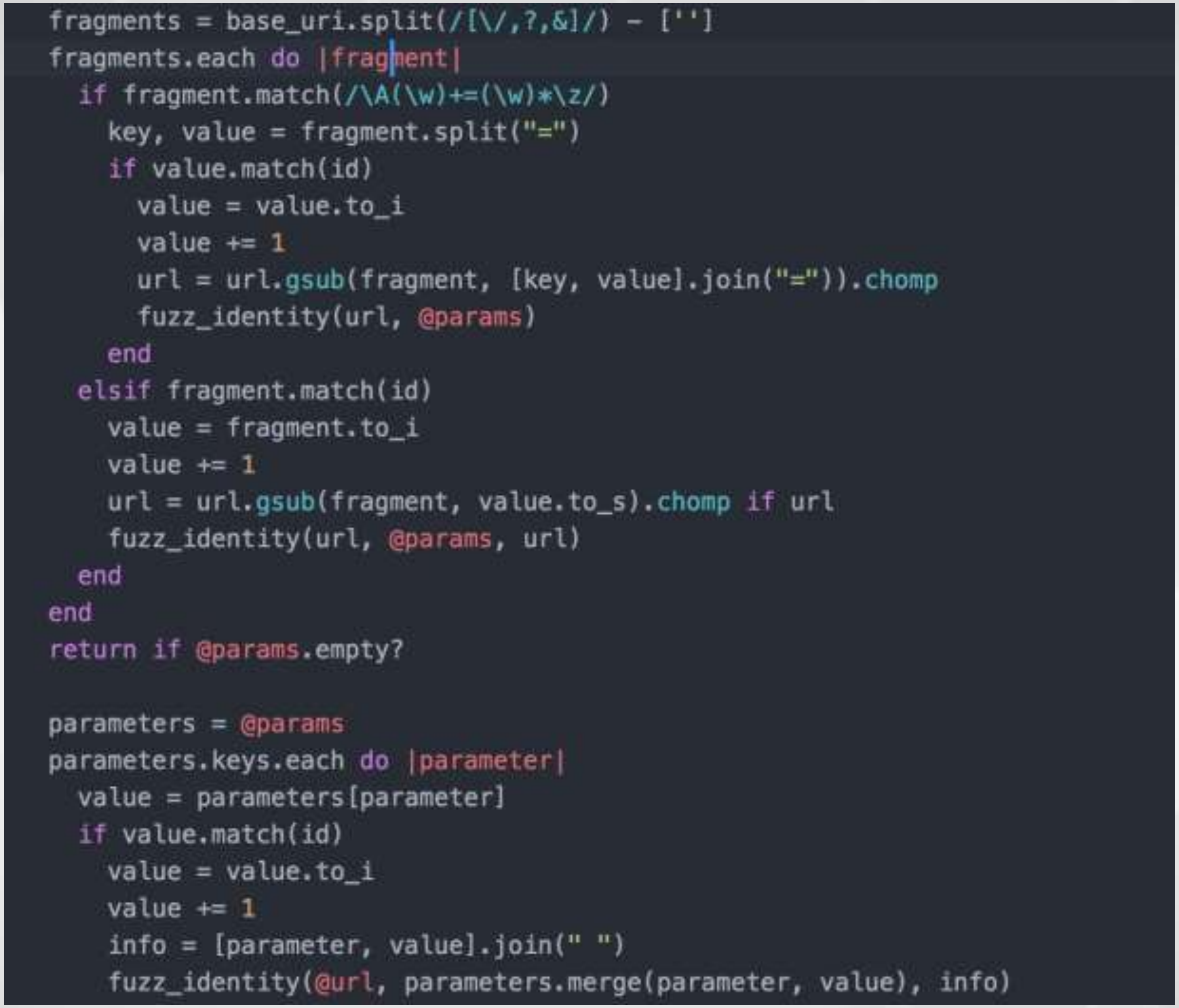


```ruby
require 'API_Fuzzer/vulnerability'
require 'API_Fuzzer/error'
require 'API_Fuzzer/request'

module API_Fuzzer
  class XxeCheck

    def self.scan(options = {})
      @url = options[:url] || nil
      @params = options[:params]
      @scan_hash = options[:scan]
      @cookies = options[:cookies] || {}
      @headers = options[:headers] || {}
      fuzz_xml_params
    end
```



```ruby
class PingController < ActionController::Base
  def index
    @scan = Scan.find(params[:id])
    @scan.vulnerabilities.create!(
      status: 'HIGH',
      class_type: 'Vulnerability',
      description: "Possible XXE vulnerability in #{@scan.url}",
      value: body
    ) if @scan
    render json: { status: :ok }
  end
```

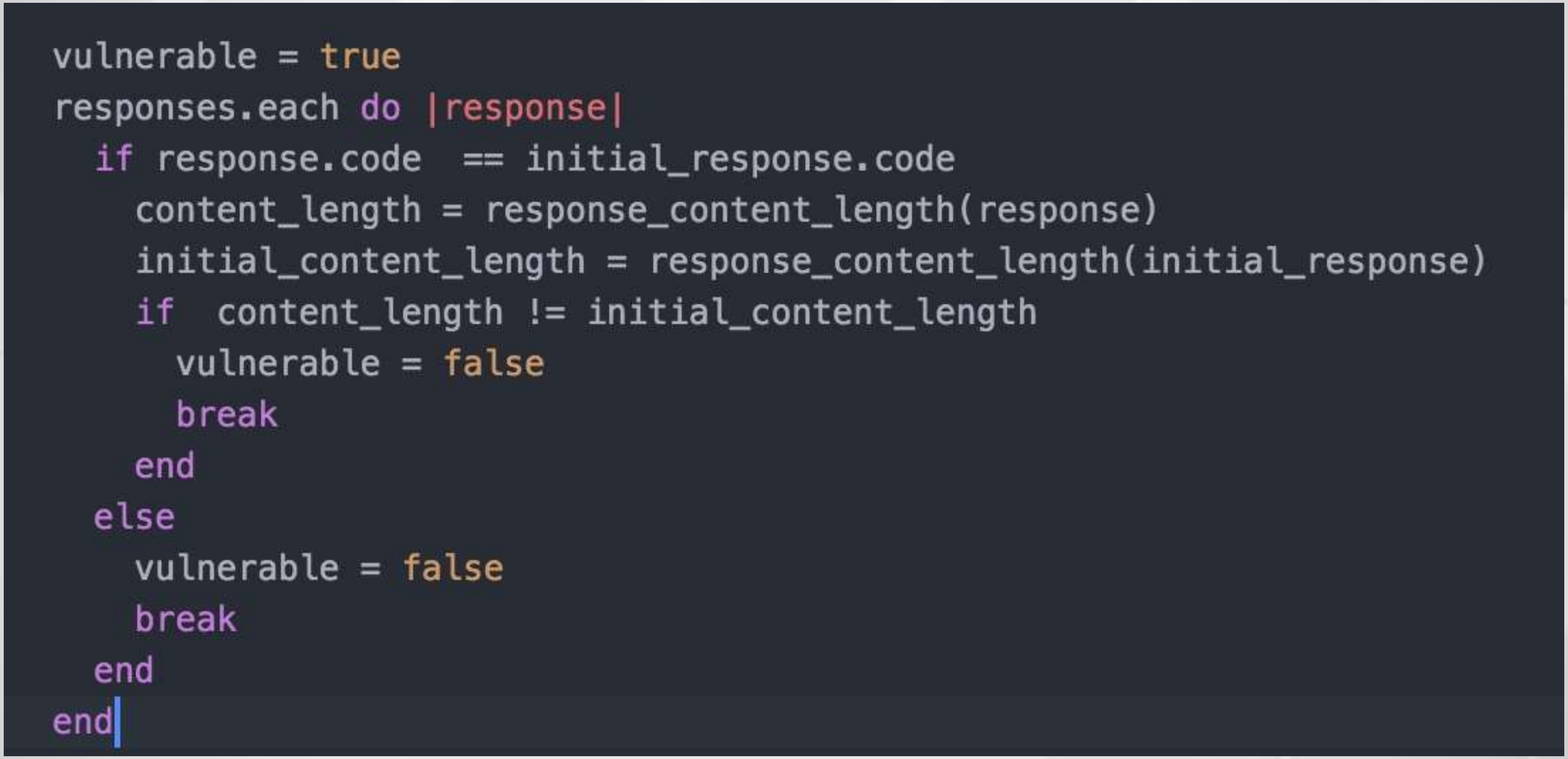
Fuzzapi sample approach for Privilege Escalation



```
fragments = base_uri.split(/[\/,?,&]/) - ['']
fragments.each do |fragment|
  if fragment.match(/\A(\w)+=(\w)*\z/)
    key, value = fragment.split("=")
    if value.match(id)
      value = value.to_i
      value += 1
      url = url.gsub(fragment, [key, value].join("=")).chomp
      fuzz_identity(url, @params)
    end
  elsif fragment.match(id)
    value = fragment.to_i
    value += 1
    url = url.gsub(fragment, value.to_s).chomp if url
    fuzz_identity(url, @params, url)
  end
end
return if @params.empty?

parameters = @params
parameters.keys.each do |parameter|
  value = parameters[parameter]
  if value.match(id)
    value = value.to_i
    value += 1
    info = [parameter, value].join(" ")
    fuzz_identity(@url, parameters.merge(parameter, value), info)
```

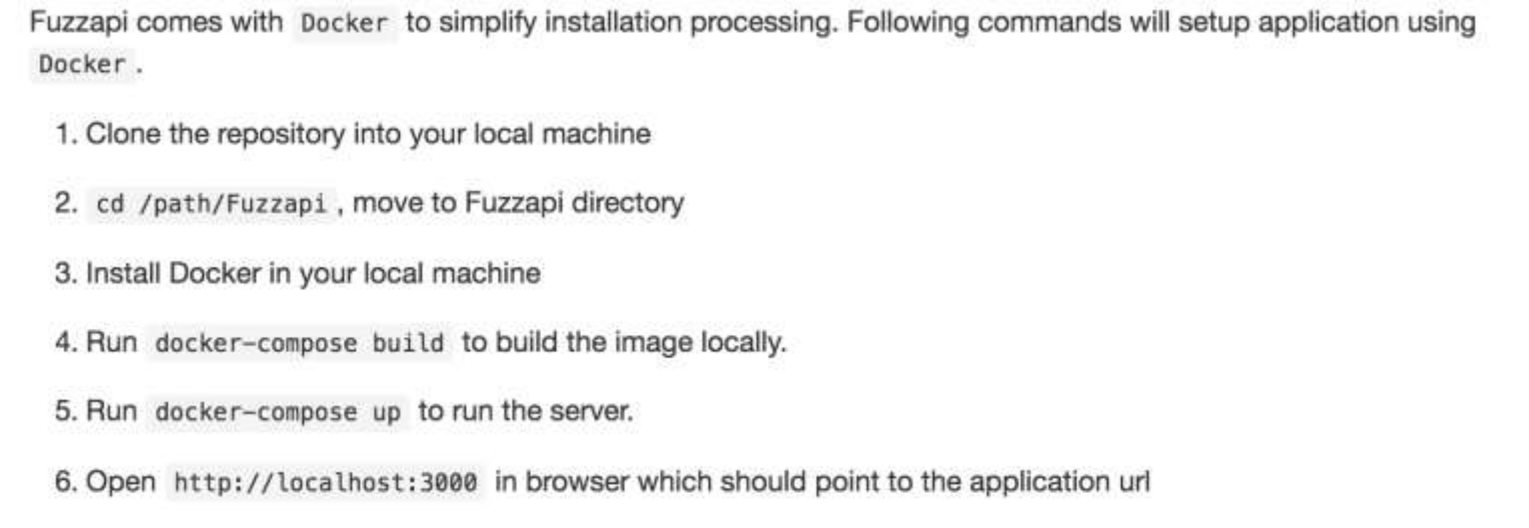
Fuzzapi sample approach for Rate limiting

- Fuzzapi sends multiple sample requests and waits for timeout/error
- Failure in limiting requests allows to perform this check



```
vulnerable = true
responses.each do |response|
  if response.code  == initial_response.code
    content_length = response_content_length(response)
    initial_content_length = response_content_length(initial_response)
    if  content_length != initial_content_length
      vulnerable = false
      break
    end
  else
    vulnerable = false
    break
  end
end
```

Docker :D :D \m/

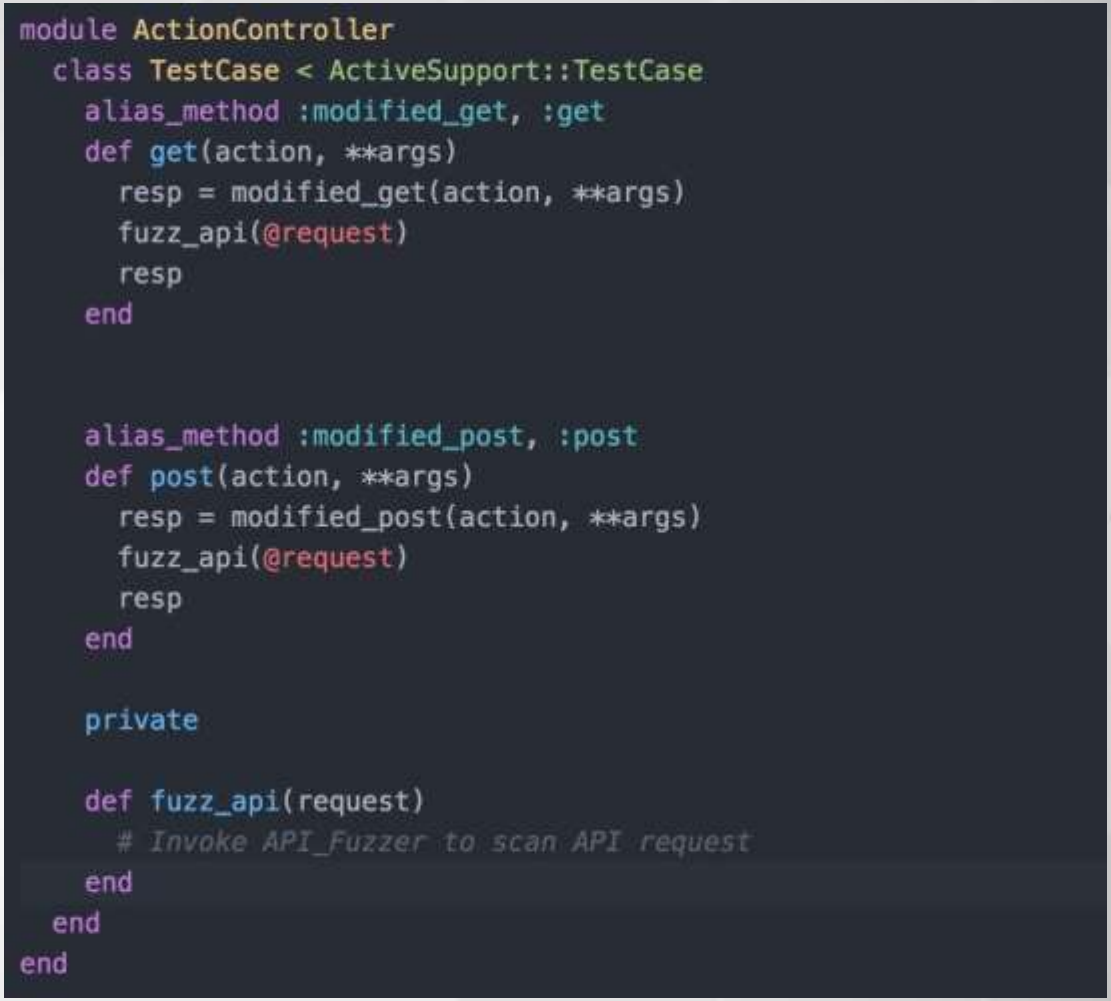


Fuzzapi comes with `Docker` to simplify installation processing. Following commands will setup application using `Docker`.

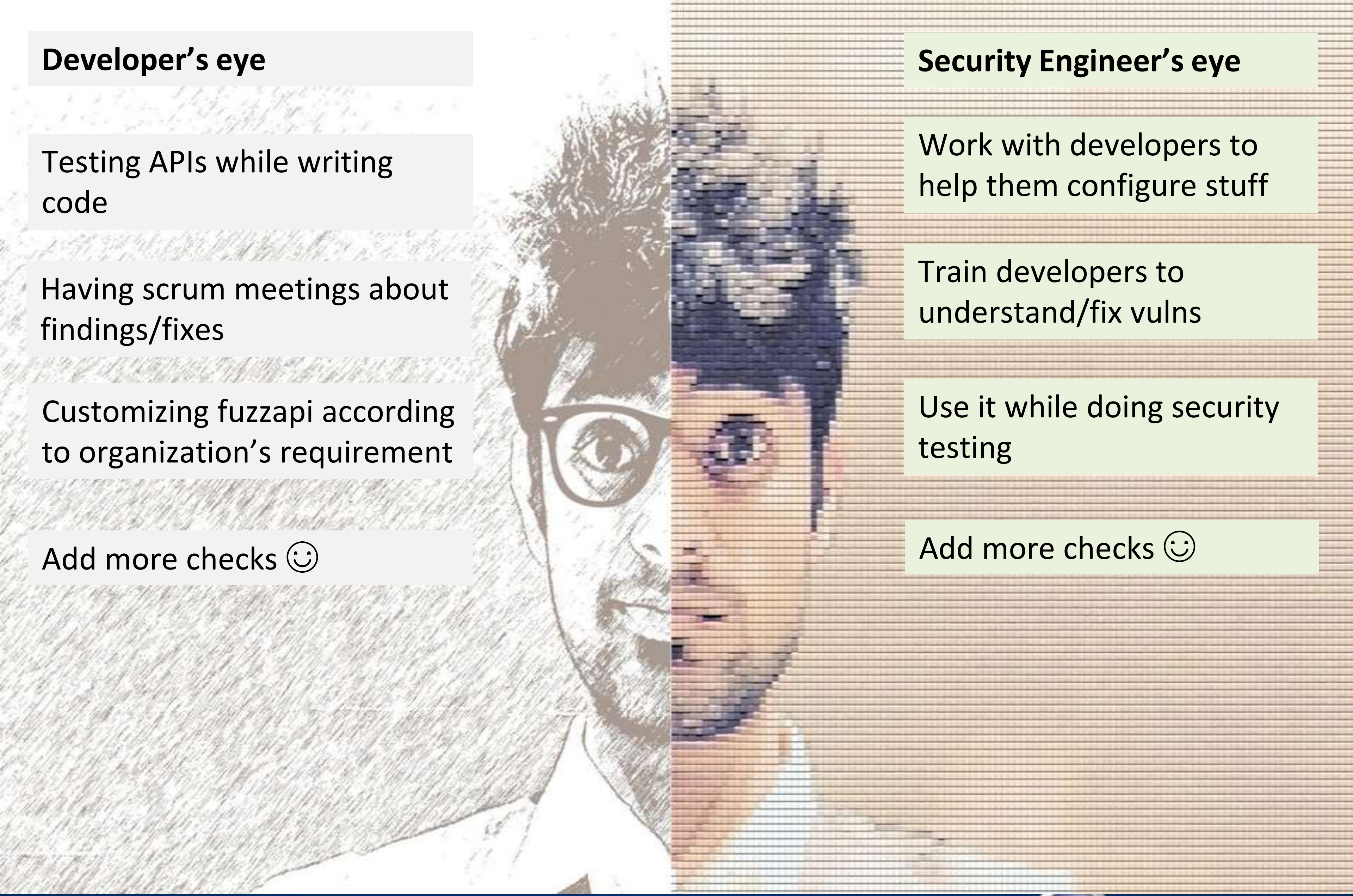
1. Clone the repository into your local machine
2. `cd /path/Fuzzapi`, move to Fuzzapi directory
3. Install Docker in your local machine
4. Run `docker-compose build` to build the image locally.
5. Run `docker-compose up` to run the server.
6. Open `http://localhost:3000` in browser which should point to the application url

Continuous integration --Rails !!!

- Identify test requests
- Use API_Fuzzer module with test request
- Run scans



```ruby
module ActionController
  class TestCase < ActiveSupport::TestCase
    alias_method :modified_get, :get
    def get(action, **args)
      resp = modified_get(action, **args)
      fuzz_api(@request)
      resp
    end


    alias_method :modified_post, :post
    def post(action, **args)
      resp = modified_post(action, **args)
      fuzz_api(@request)
      resp
    end

    private

    def fuzz_api(request)
      # Invoke API_Fuzzer to scan API request
    end
  end
end
```



Developer’s eye

Testing APIs while writing code

Having scrum meetings about findings/fixes

Customizing fuzzapi according to organization’s requirement

Add more checks ☺

Security Engineer’s eye

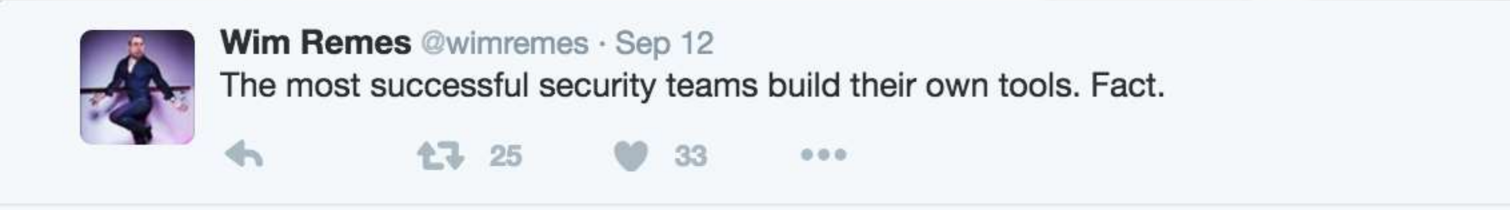
Work with developers to help them configure stuff

Train developers to understand/fix vulns

Use it while doing security testing

Add more checks ☺

CONNECT LEARN GROW



Wim Remes @wimremes · Sep 12

The most successful security teams build their own tools. Fact.

Roadmap for fuzzapi/us

Add more checks

Write more blogs

Make more tutorial videos

Write more tools

Repeat

Oh yea btw :D Don’t you want links to download?

API_Fuzzer gem: https://github.com/lalithr95/API-fuzzer

fuzzapi: https://github.com/lalithr95/Fuzzapi

For queries/concerns/feedback/rant:
Twitter:
@abhijeth
@lalithr95
@srini0x00

It’s 2016 and if you still don’t know about bug bounties/responsible disclosures, you should say hi to these guys

@Bugcrowd

@synack

@Hacker0x01

Thanks ☺

and all the security folks for contributing to the open source community ☺