Subdomain takeover, the use after free of the internet

HIP Berlin 2022

Emile Hansmaennel 2022-12-28

$whoami

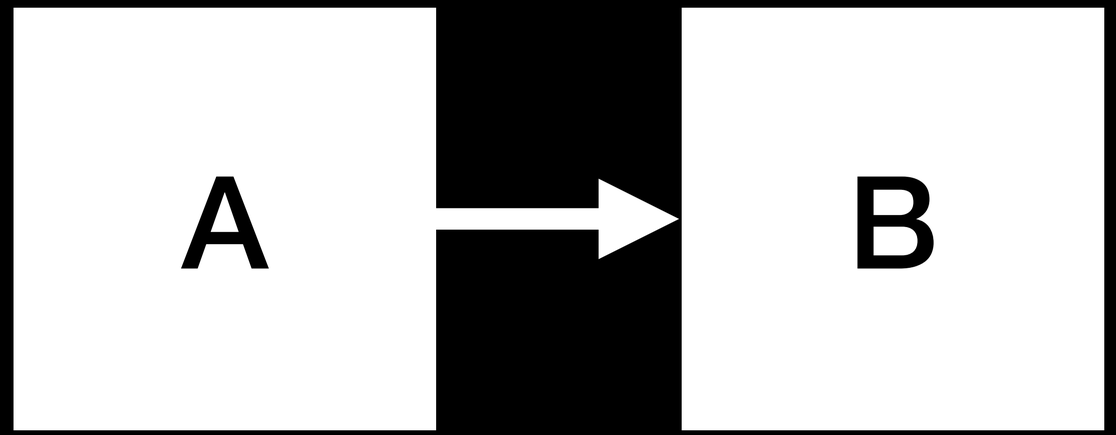
- Emile / @hanemile@chaos.social / https://emile.space
- Daytime
  - Pentesting
- Nighttime
  - Foo @chaosdorf (Düsseldorf)
  - CTF @ALLES! && @Sauercl0ud

Use-After-Free

In general…

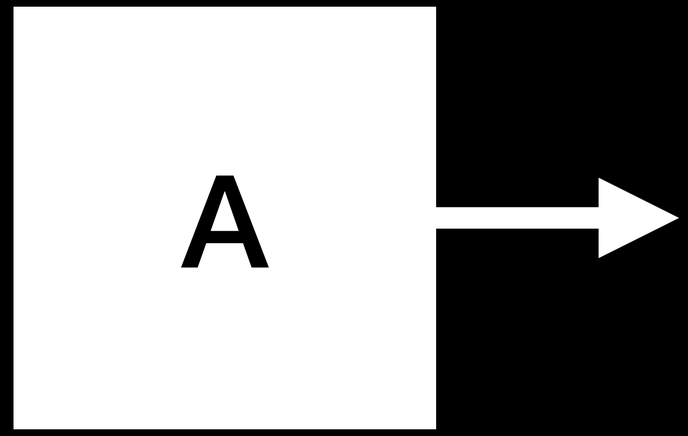
Use-After-Free

In general



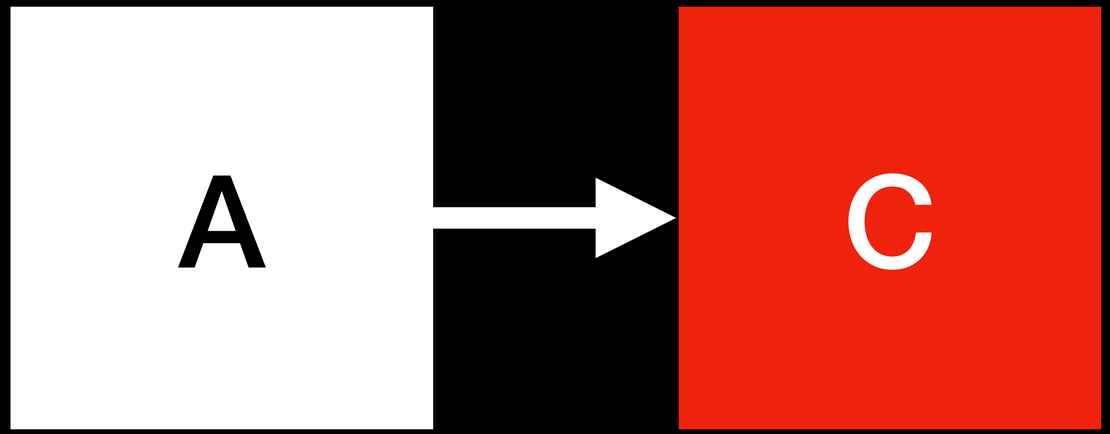
Use-After-Free

In general



Use-After-Free

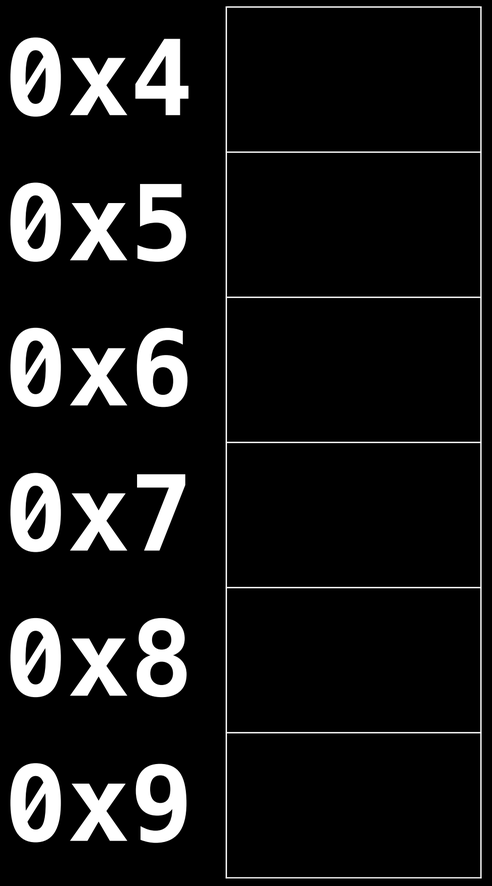
In general



Use-After-Free (Memory allocator)

Examples

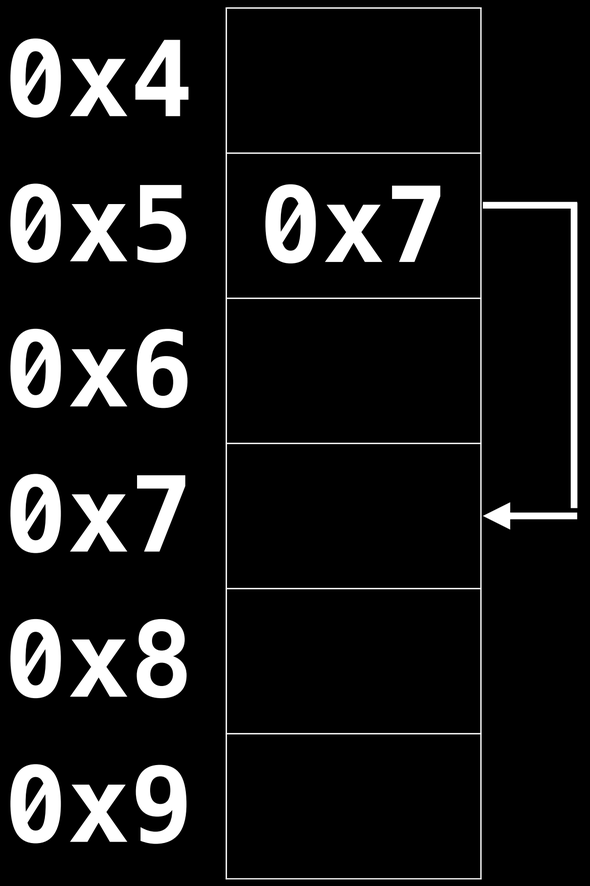
- Initial state



Use-After-Free (Memory allocator)

Examples

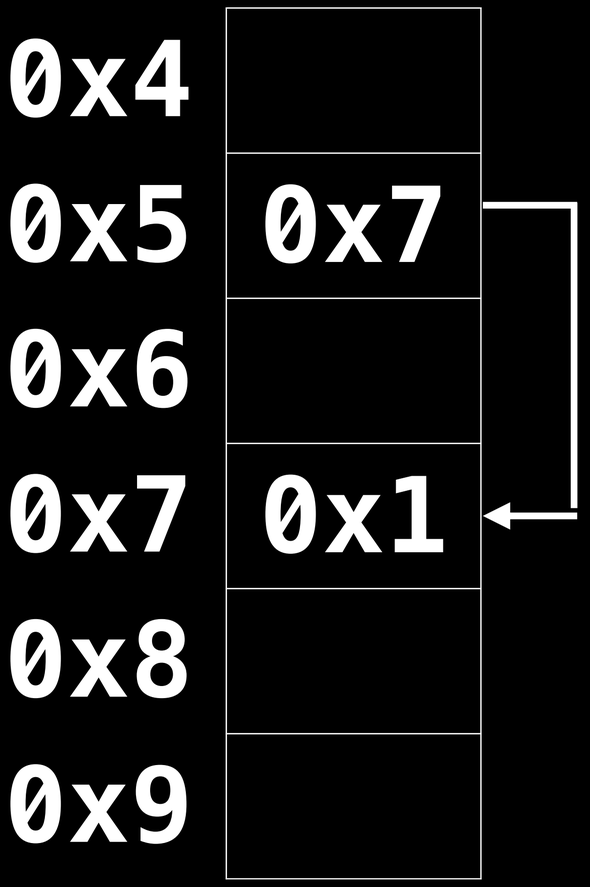
- Initial state
- allocate



Use-After-Free (Memory allocator)

Examples

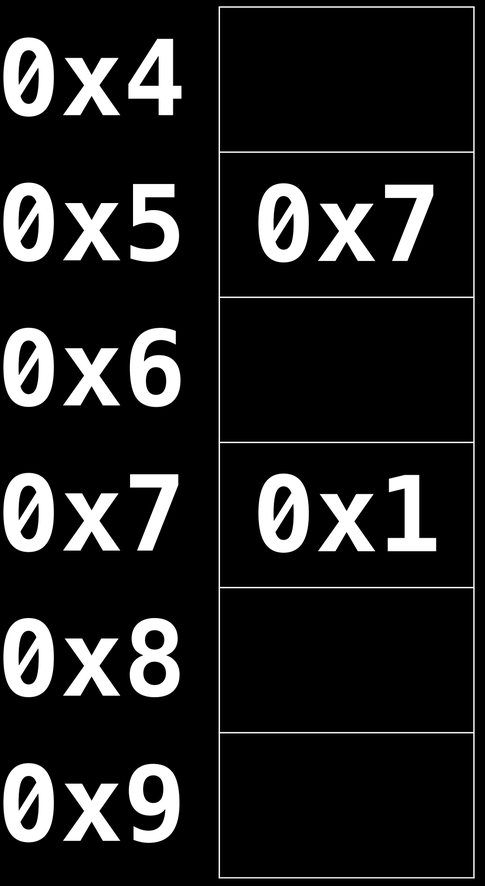
- Initial state
- allocate
- use



Use-After-Free (Memory allocator)

Examples

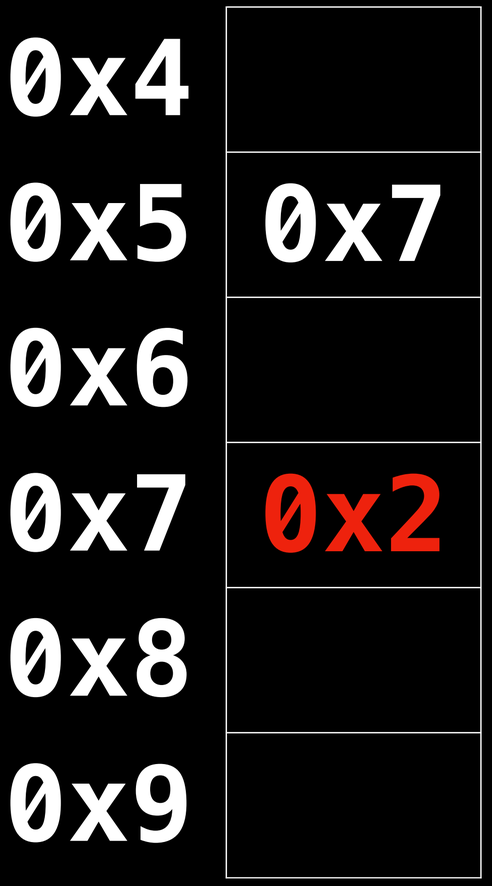
- Initial state
- allocate
- use
- free



Use-After-Free (Memory allocator)

Examples

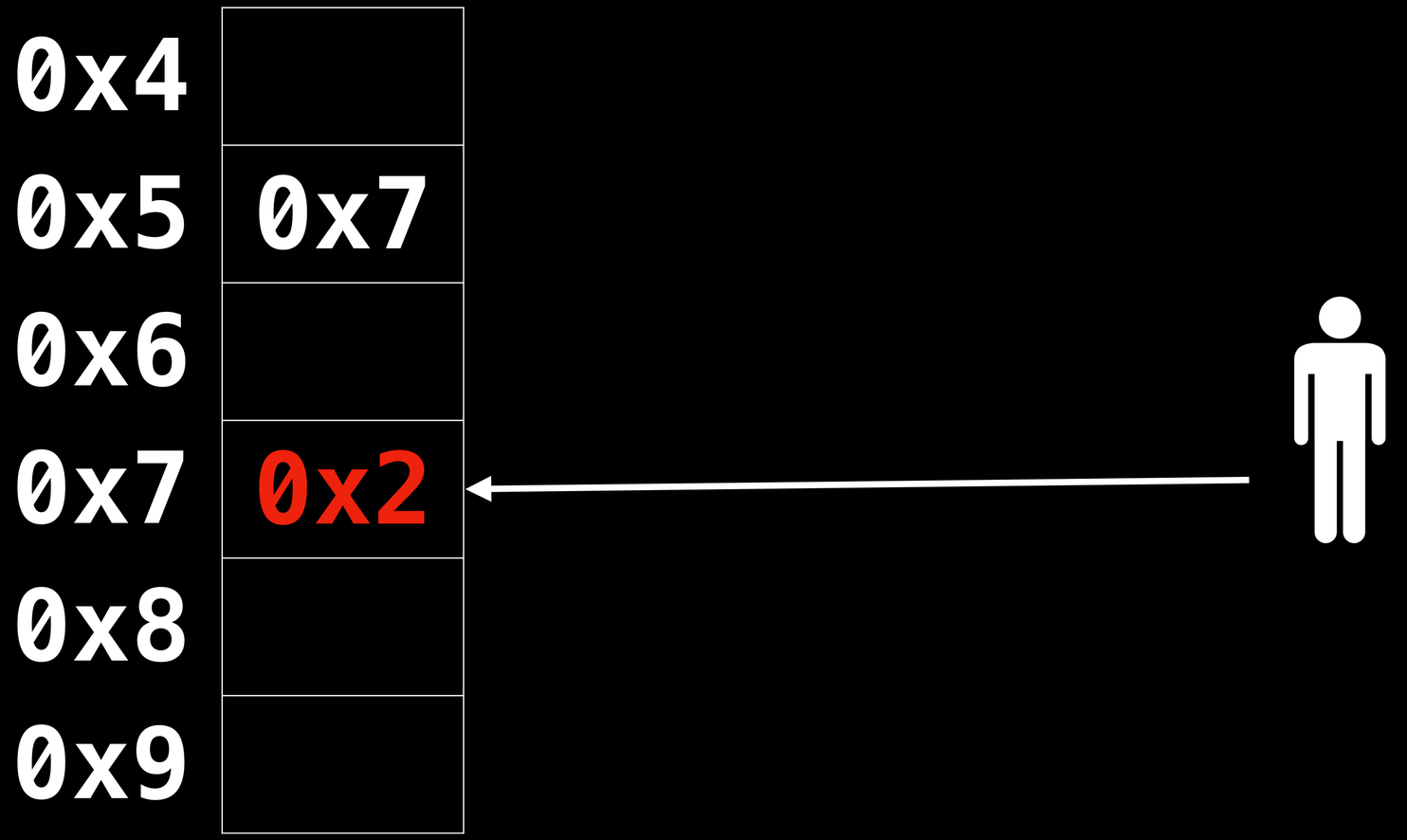
- Initial state
- allocate
- use
- free



Use-After-Free (Memory allocator)

Examples

- Initial state
- allocate
- use
- free
- use-after-free



Storytime: Out of scope

Subdomain takeover example

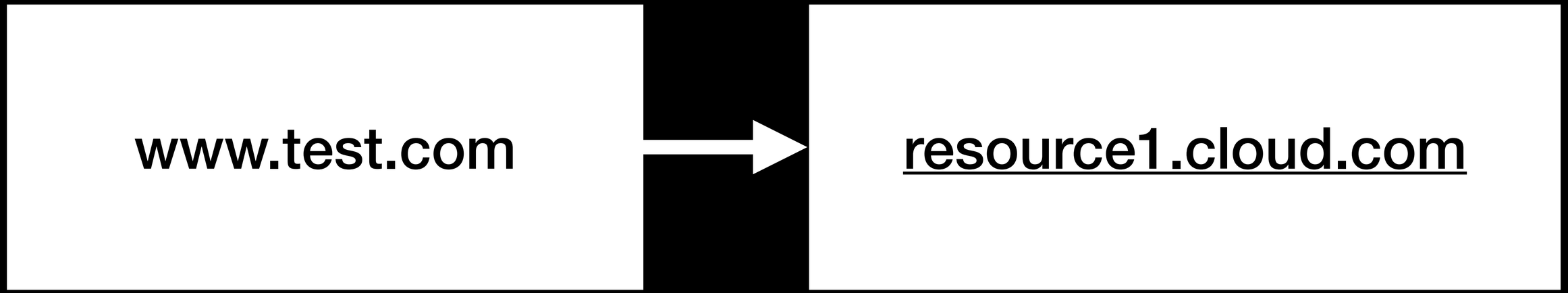
Subdomain takeover example

A small recap

A	abc.example.com	→	123.231.132.213
CNAME	def.example.com	→	resource1.cloud.com

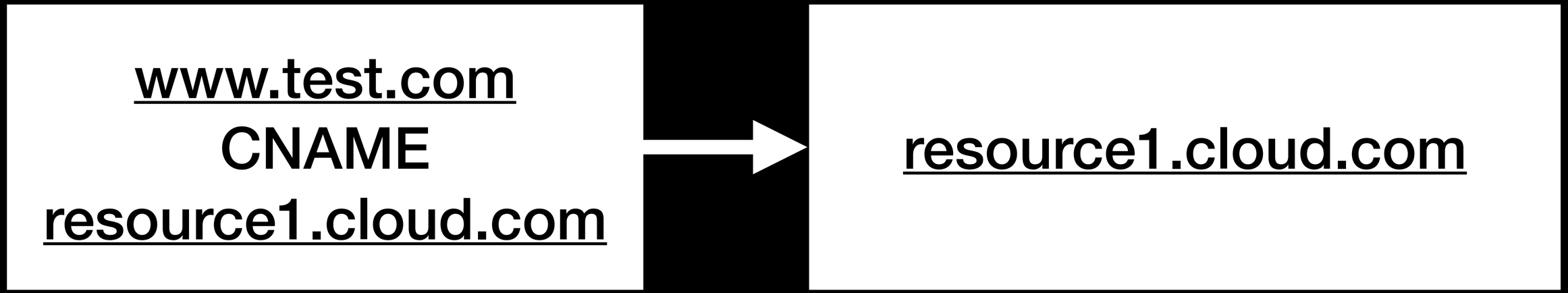
Subdomain takeover example

A small recap



Subdomain takeover example

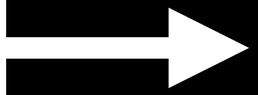
A small recap



Subdomain takeover example

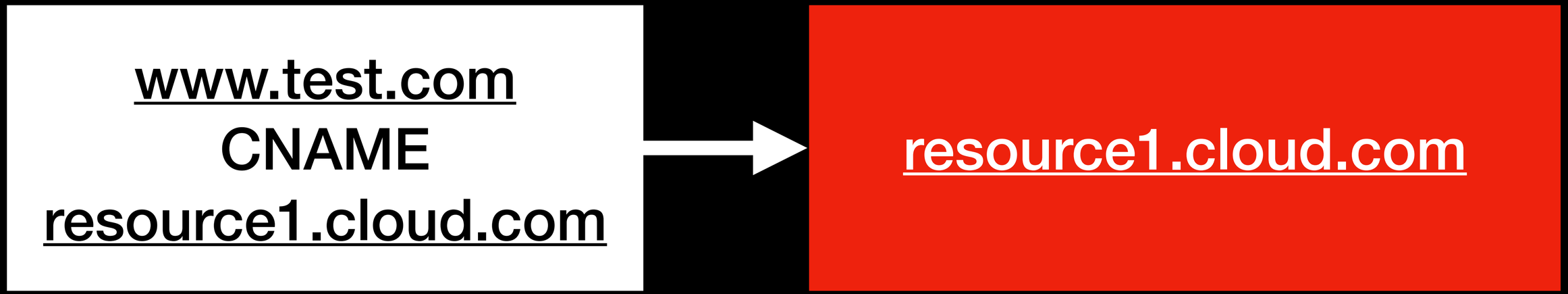
A small recap

www.test.com
CNAME
resource1.cloud.com



Subdomain takeover example

A small recap



Automation

Find Targets

- https://github.com/edoverflow/can-i-take-over-xyz

Process

- List of domains
- Get Subdomains
- Check CNAME records
- Check if target of pointer is “allocatable”
- Verify
- Report

Bash, a tool for building “highly sophisticated automation pipelines”

```
while read line; do curl -s https://crt.sh/\?q\=%.
$line\&output\=json | jq ".[].name_value" | xargs echo
-e | sed "s| |\n|g" | sed "s|^\*.*$||" | sort | uniq |
xargs -I {} -P 16 -n 1 bash -c 'dig -t A {} | tee -a
cname.txt | rg -o "CNAME.*" | sed "s|CNAME||g" | sed "s|
\.$||g" | tr -d "\t "' | tee -a dig_out.txt | xargs -P
16 -n 1 -I {} curl -m 3 -s -w "%{http_code} {}\n" -o /
dev/null {} | rg "^000" | sed "s|^000||" | xargs -I {}
rg "CNAME.*{}" cname.txt ...
```

Automation

Bash, a tool for building “highly sophisticated automation pipelines”

```
while read line; do curl -s https://crt.sh/\?q\=%.$line\&output\=json
    | jq ".[].name_value"
    | xargs echo -e
    | sed "s| |\n|g"
    | sed "s|^\*.*$||"
    | sort
    | uniq
    | xargs -I {} -P 16 -n 1 bash -c '
      dig -t A {}
      | tee -a cname.txt
      | rg -o "CNAME.*"
      | sed "s|CNAME||g"
      | sed "s|\.$||g"
      | tr -d "\t "'
    | tee -a dig_out.txt
    | xargs -P 16 -n 1 -I {} curl -m 3 -s -w "%{http_code} {}\n" -o /dev/null {}
    | rg "^000"
    | sed "s|^000||"
    | xargs -I {} rg "CNAME.*{}" cname.txt ...
```

Step by Step

Status: Nothing

- get all subdomains for target (crt.sh)
- parse the output (crt.sh: `&output=json`)
- `sort` + `uniq`

- new status: we've got a list of subdomains!

Step by Step

Status: List of Subdomains

- `dig -t A {} > dig.txt`
- extract all CNAME (`rg -o "CNAME" {}`)

- new status: list of all subdomains with CNAME records somewhere!

Step by Step

Status: List of all subdomains with CNAME records somewhere!

- parse output (`sed` + `awk` + `rg` = ❤️)
- make HTTP request to all subdomains with CNAME record
- filter HTTP 000

- new status: list of all subdomains with CNAME records with a dead target!

Step by Step
Status: Validate

- Create the “missing” resource
  - [a-zA-Z0-9].<something>
  - [a-zA-Z0-9].<region>.<something>

- azure cli (az)

Verification

- Create resource
- Access resource via CNAME record
- Check if resource can be accessed

Fix it!

Fix it!

Single case

- remove the CNAME record
- implement measures that make sure that this doesn’t happen again

Fix it!
META

- Before problems arise:
  - don’t allow reallocation of already used subdomains
    - “GitHub problem”
  - restrict deterministic subdomain allocation
    - numbers (subdomain12387512.cloud.com)

Conclusion

Stats
(1 weekend)

- 75 Companies
- 309+ validated takeovers (azure only)

Reporting
(horrendous)

- 48 of 75 with twitter (64%)
  - 9 replies
  - 5 DMs
- 0 of 75 with a security.txt

Further doings

- Look into where resources are allocated
- Simmilar problems may be present elsewhere
- Think of some way to define these actions in a more formalized way for automated identification of these problems

use-after-free can be present in places you might not expect

try to find solutions at scale

THE END

$contact

- @hanemile everywhere
- mail: subdomaintakeover@emile.space
- mastodon: @hanemile@chaos.social
- matrix: @hanemile:matrix.org