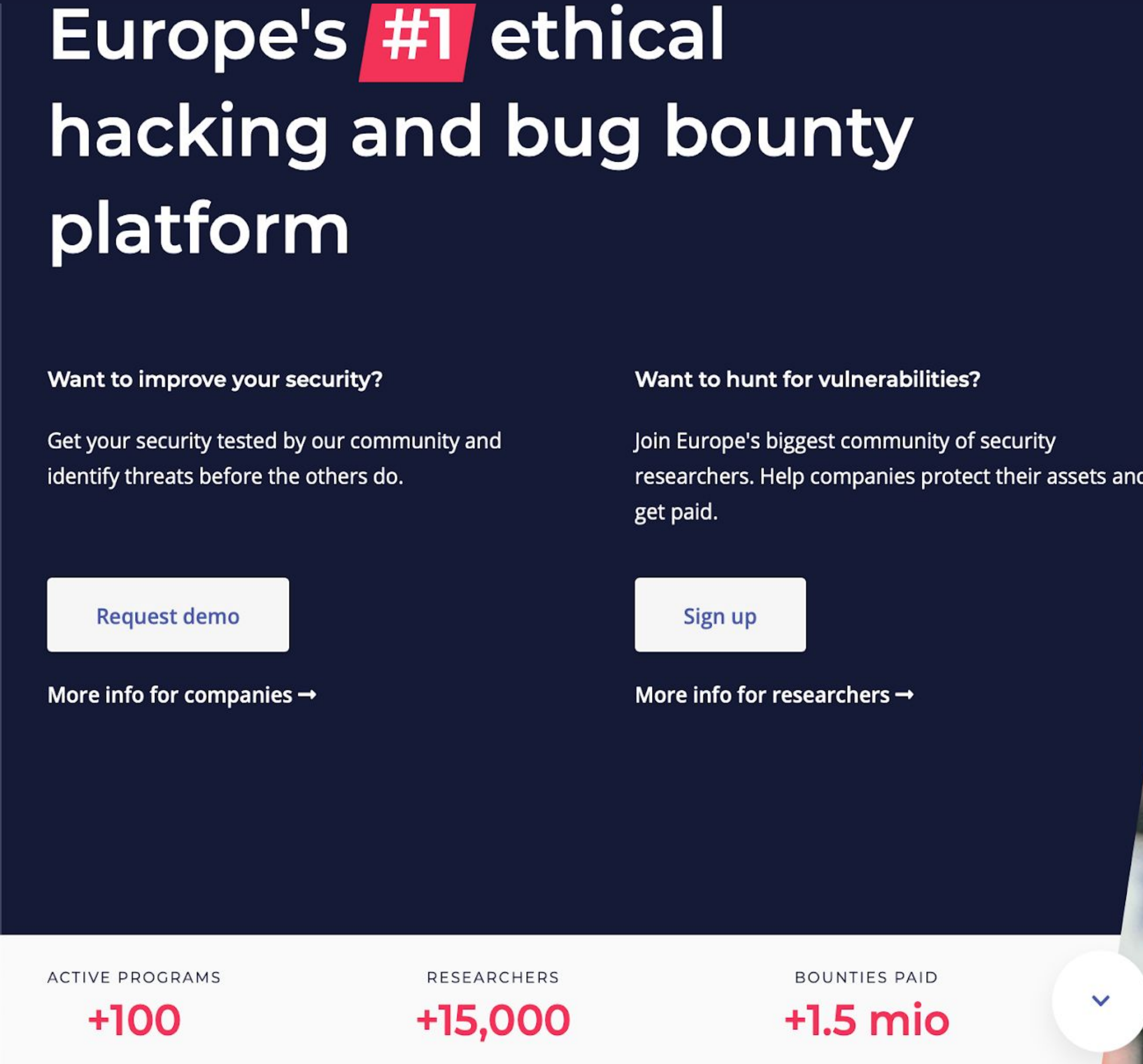
APISEC

an Intigriti look

INTIGRITI
ETHICAL HACKING PLATFORM

1. Intigriti



Europe's #1 ethical hacking and bug bounty platform

Want to improve your security?

Get your security tested by our community and identify threats before the others do.

Want to hunt for vulnerabilities?

Join Europe's biggest community of security researchers. Help companies protect their assets and get paid.

ACTIVE PROGRAMS +100

RESEARCHERS +15,000

BOUNTIES PAID +1.5 mio

Niels Hofmans

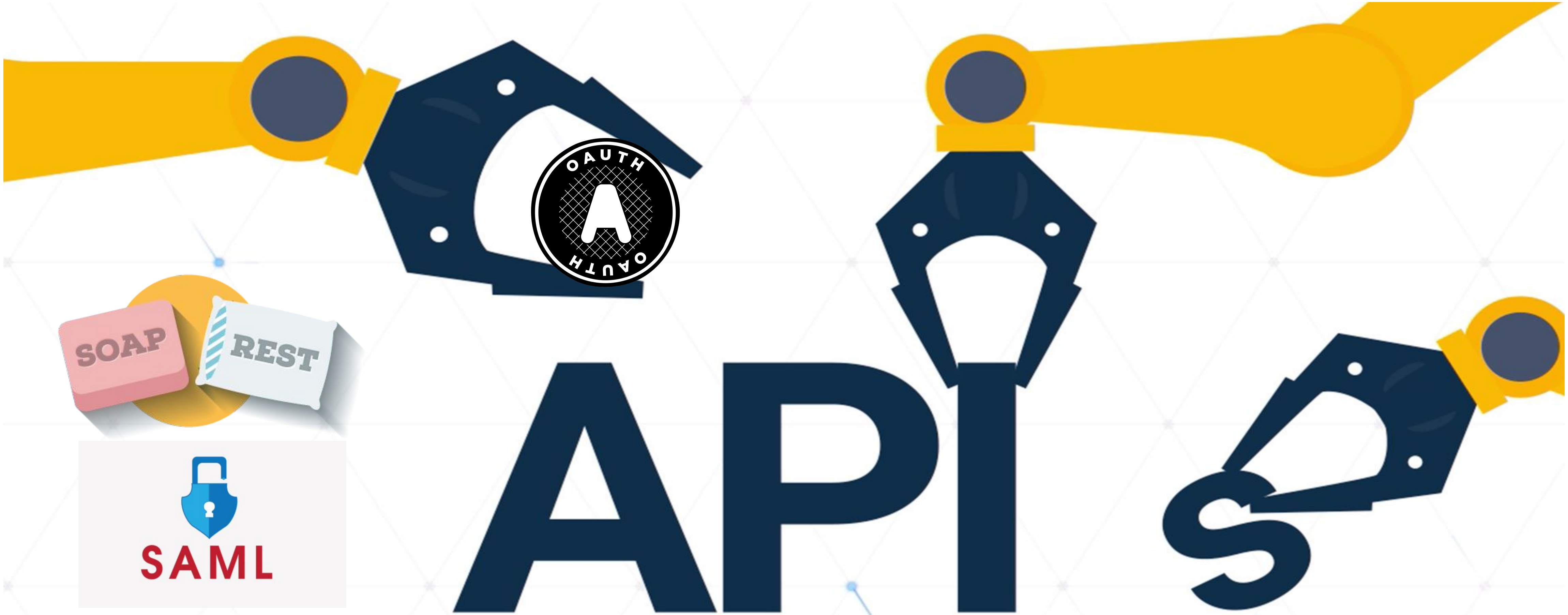
niels@intigriti.com
https://intigriti.com

Security & Innovation

2. Bug Bounty

- Transparent ethical **hacking**
- **Companies** create scoped **programs**
  Responsible Disclosure, Private, Dedicated, Public
- Researchers report **vulnerabilities**
- Submissions are **triaged** by us
- **Bounties** are based **on impact**
- Security **coaching**

3. Testing Methodologies



APIs start out small … but easily get more and more complex.

3. Testing Methodologies

- **Black box testing**
  investigate a locked chest (crawl the application)

- **Gray box testing**
  investigate a locked chest, but with clues (specify critical calls)

- **White box testing**
  hand over the chest schematics (Swagger, OpenAPI, ...)

3. Testing Methodologies

Methodology	Input	Output	Focus	Tooling	Value
Black box	Scope	Vulnerabilities	Vulnerabilities	Scanning Proxy	Security Compliance
Gray box	Scope Clues	Vulnerabilities Pain points		Scanning Proxy Scripts	
White box	Scope Clues Specs	Vulnerabilities Pain points Flaws	Vulnerabilities Processes	Scanning Proxy Scripts Validators	Security Compliance Business

3. Testing Methodologies

- **Whitebox** testing provides most added value

- Security **requirements**
  Divide into:
  - Baseline framework
  - Project requirements
  - Security controls

- Automated **tooling**

- **Abuse** cases

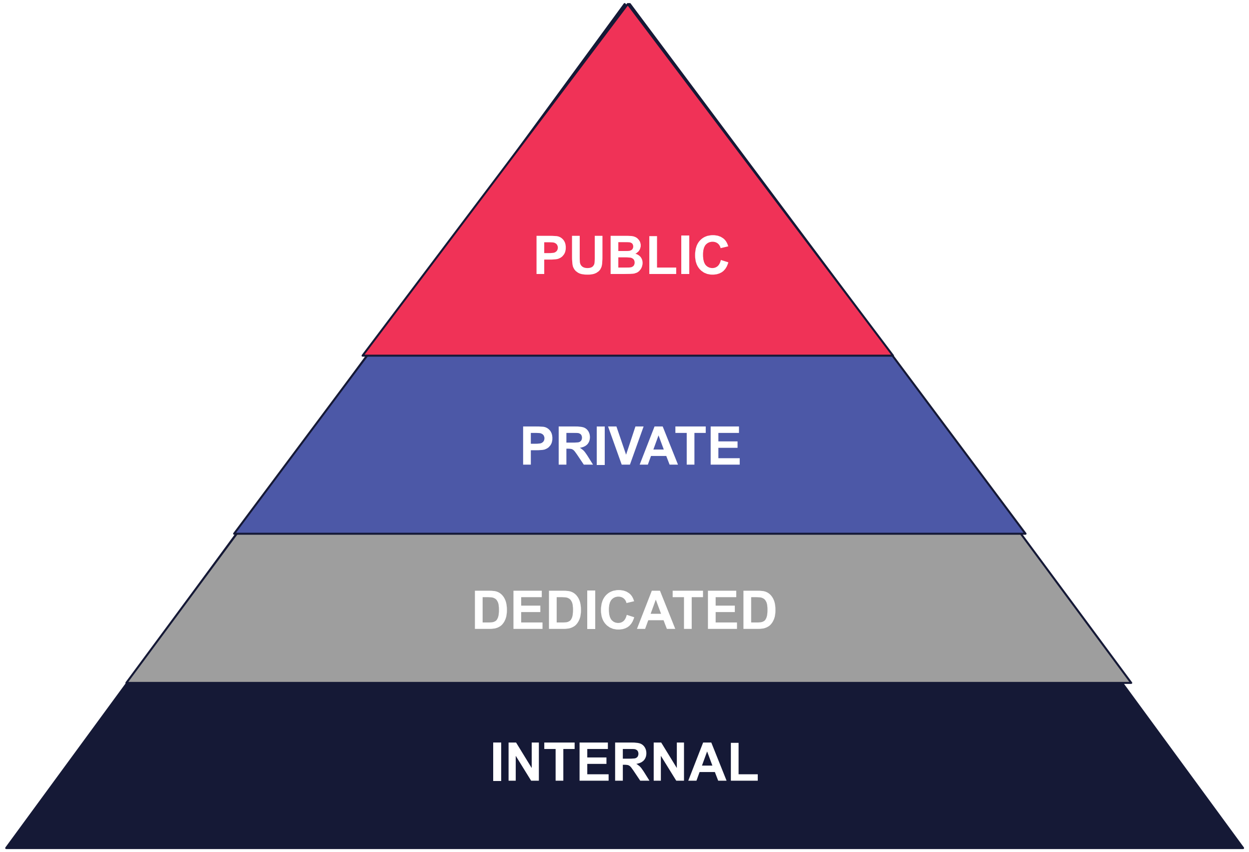
- **Manual** testing

3. Testing Methodologies

The Intigriti **Testing Triangle**

- + **confidence**
- + **creativity**
- + **maturity**

- - **cost**
- - **involvement**
- - **feature focus**



4. Intigriti API Top 5

1. **Business logic flaws**
2. **API XSS**
3. **Outdated software**
4. **Information/PII leakage**
5. **Broken access control**

4. Intigriti API Top 5

1. **Business logic flaws**

Insecure Direct Object Reference

PII Leakage

+ Never trust user input
+ UUIDs or tokens > numbers

Example

GET /api/v1/users?id=2428

GET /api/v1/users?id=2429

4. Intigriti API Top 5

2. API XSS

Missing context aware encoding

- + Always provide Content-Type
- + Output encoding on API & FE

Example

GET /api/v1/users?id=2428

Content-Type: application/json

{“name”:”<script>alert(1)</script>”}

GET /system/status/user/23

Content-Type: text/plain

<script>alert(1)</script>

4. Intigriti API Top 5

3. Outdated software

Patch policies for complex stacks

+ Infrastructure visibility
+ Patch policies

Example

HTTP Headers

File Metadata (PDF, ...)

Fingerprinting (Wordpress, ...)

...

4. Intigriti API Top 5

4. Information/PII leakage

Disclosing secrets

+ proper secret management

+ scan for secrets during deploy

Example

```
GET /config.php.bak
<?php
    $DB_password = “SECRET”;

GET /static/mainIndex.js
…
let devApiCredentials = “SECRET”;
…
```

4. Intigriti API Top 5

5. Broken access control

Breaking flows

- Authorize centrally

  (e.g. /admin/... requires token scope)

- Threat model functional flows

Example

POST /admin/authorize/new

Cookie: session=regularUser

user=regularUser

5. Lessons Learned

1. **Get your requirements right from the start**
   OWASP ASVS, OWASP SAMM

2. **Whitebox testing**
   Negative test cases, automated tooling

3. **No harm in hacking ;-)**
   Dedicated + Bug Bounty

That’s all folks

APISEC by Intigriti

niels@intigriti.com

https://intigriti.com

INTIGRITI
ETHICAL HACKING PLATFORM