SCANTIST

Docker and SCA

Overview

SCA for Docker

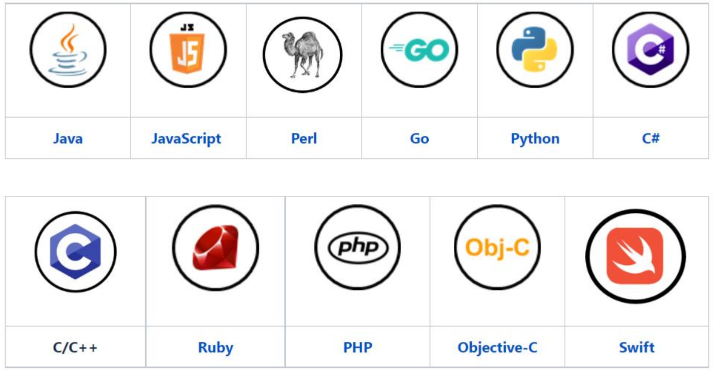
- SCA scan for docker images
- challenges
- effort to improve

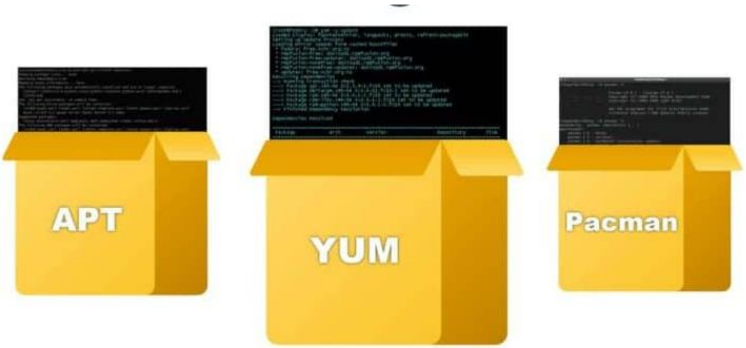
Docker for SCA

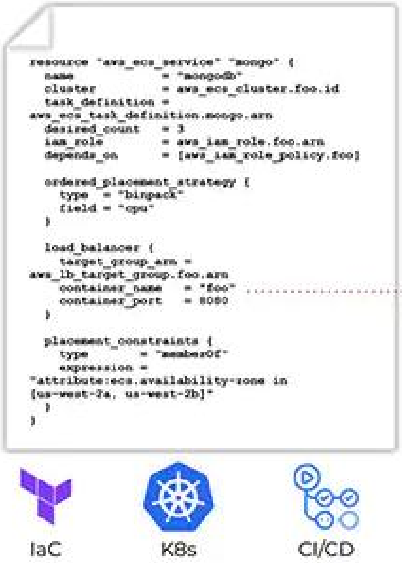
- docker as a sandbox for SCA

Disclaimer: all the figures from internet is for sharing and education usage only

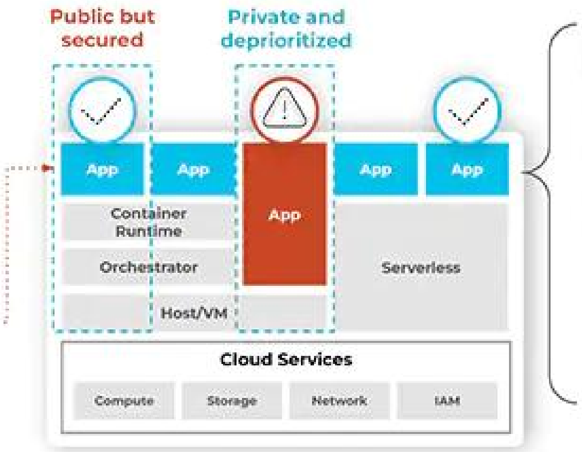
What is SCA



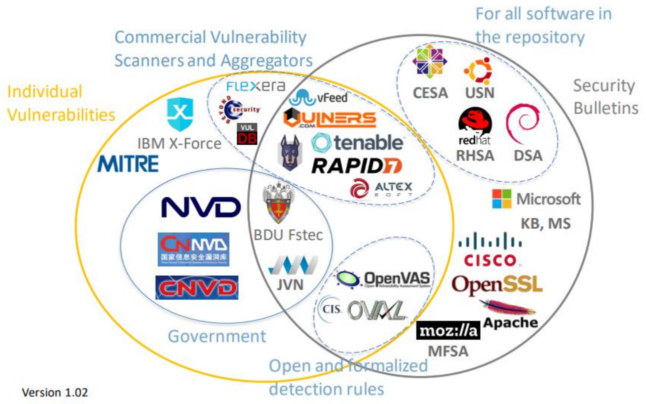


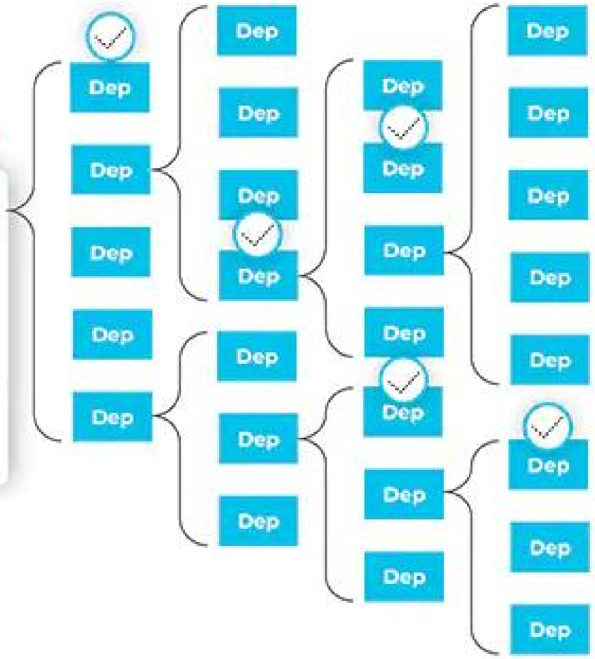


```
resource "aws_ecs_service" "mongo" {
  name            = "mongodb"
  cluster         = aws_ecs_cluster.foo.id
  task_definition =
aws_ecs_task_definition.mongo.arn
  desired_count   = 3
  iam_role        = aws_iam_role.foo.arn
  depends_on      = [aws_iam_role_policy.foo]

  ordered_placement_strategy {
    type  = "binpack"
    field = "cpu"
  }

  load_balancer {
    target_group_arn =
aws_lb_target_group.foo.arn
    container_name   = "foo"
    container_port   = 8080
  }

  placement_constraints {
    type       = "memberOf"
    expression =
"attribute:ecs.availability-zone in
[us-west-2a, us-west-2b]"
  }
}
```



Source: paloaltonetworks





Software component analysis

- scan and detect direct referenced components
- infer or detect transitive dependent components
- report vulnerabilities
- report license violations

SCA for Docker

What is SCA

- More open source packages: as of June 2022, GitHub reported having over 83 million developers and more than 200 million repositories, including at least 28 million public repositories.

- More vulnerabilities in software supply chain
  direct & transitive references

- More software supply chain attacks
  supply chain poisonings

- High volume of outbreak, examples:
  - log4j
  - heartbleed in openssl

2021 State of the Software Supply Chain report, Sonatype

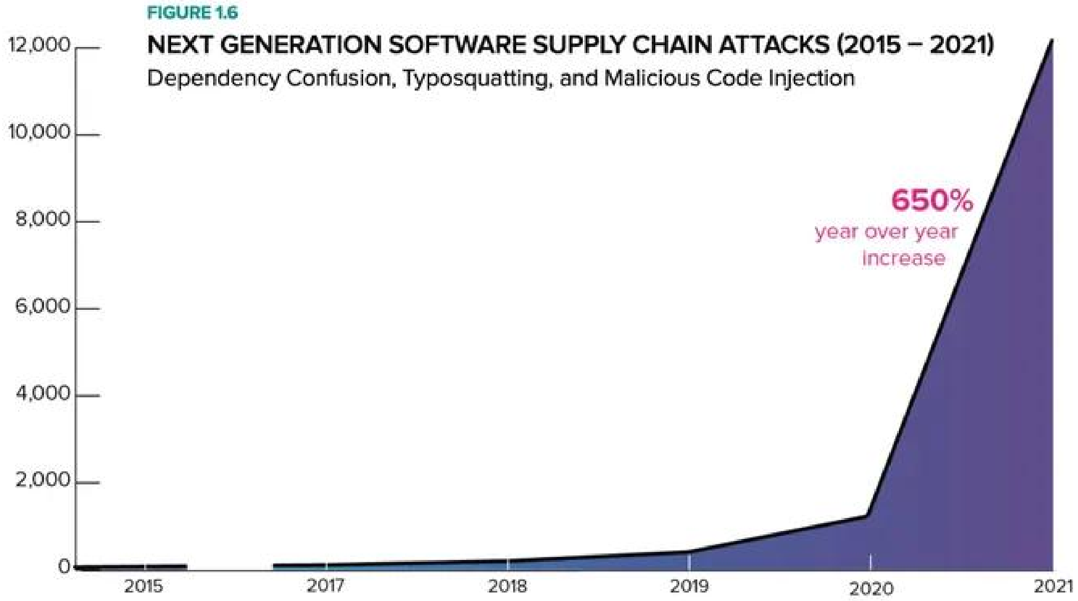
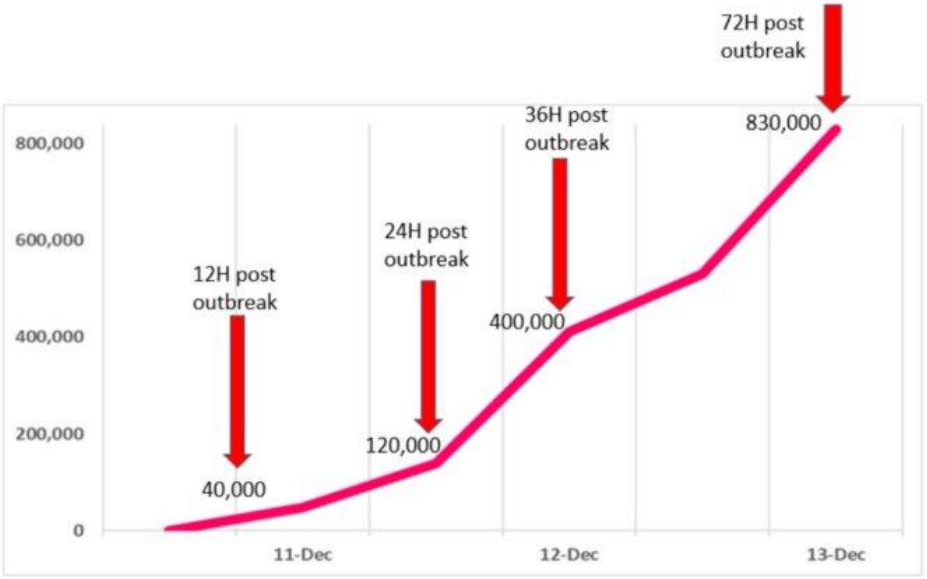


FIGURE 1.6

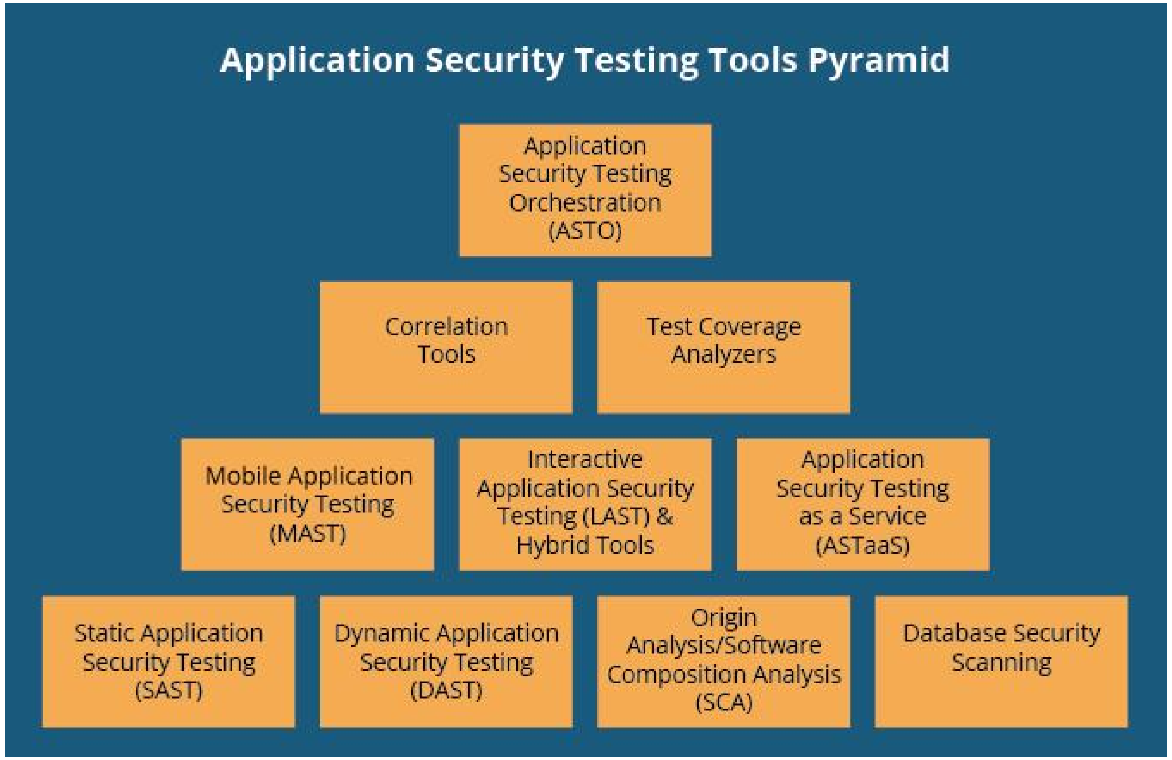
NEXT GENERATION SOFTWARE SUPPLY CHAIN ATTACKS (2015 – 2021)

Dependency Confusion, Typosquatting, and Malicious Code Injection



72 hours post initial outbreak of log4j, CVE-2021-44228

What is SCA



Application Security Testing Tools Pyramid

Application Security Testing Orchestration (ASTO)

Correlation Tools

Test Coverage Analyzers

Mobile Application Security Testing (MAST)

Interactive Application Security Testing (LAST) & Hybrid Tools

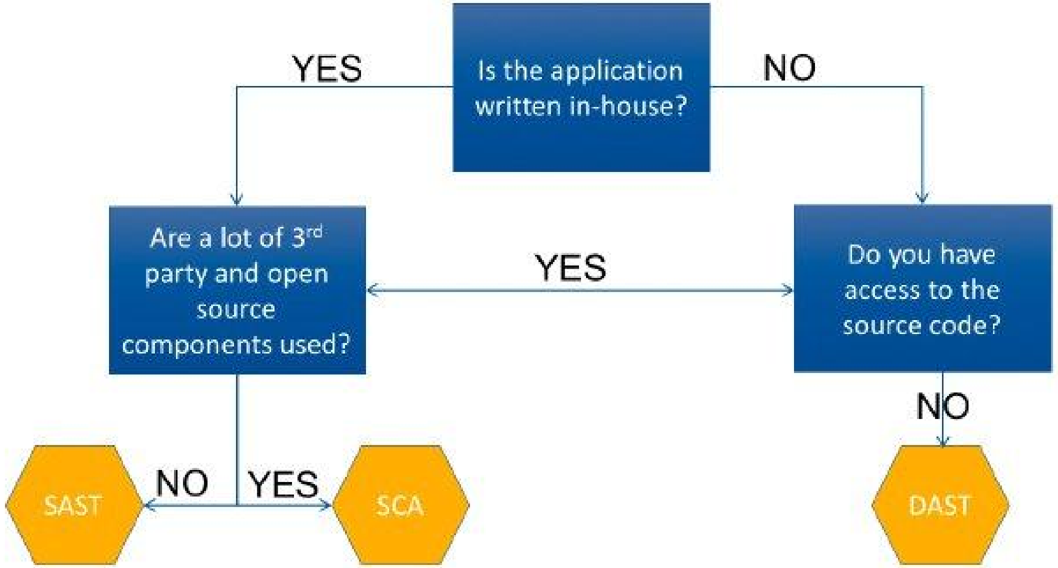
Application Security Testing as a Service (ASTaaS)

Static Application Security Testing (SAST)

Dynamic Application Security Testing (DAST)

Origin Analysis/Software Composition Analysis (SCA)

Database Security Scanning



Is the application written in-house?

YES

NO

Are a lot of 3rd party and open source components used?

YES

Do you have access to the source code?

NO

YES

NO

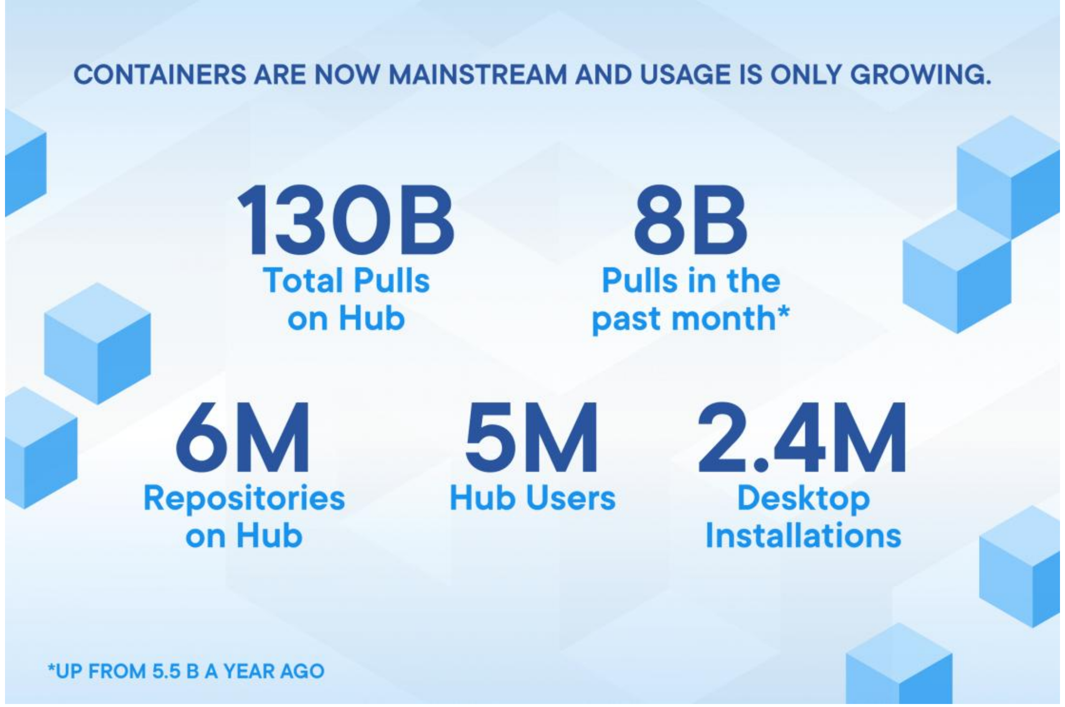
SAST

SCA

DAST

- SCA is highly effective to detect third party components and the associated **known vulnerabilities**
- SCA plays a key role in application security testing

Docker Images



CONTAINERS ARE NOW MAINSTREAM AND USAGE IS ONLY GROWING.

130B Total Pulls on Hub

8B Pulls in the past month*

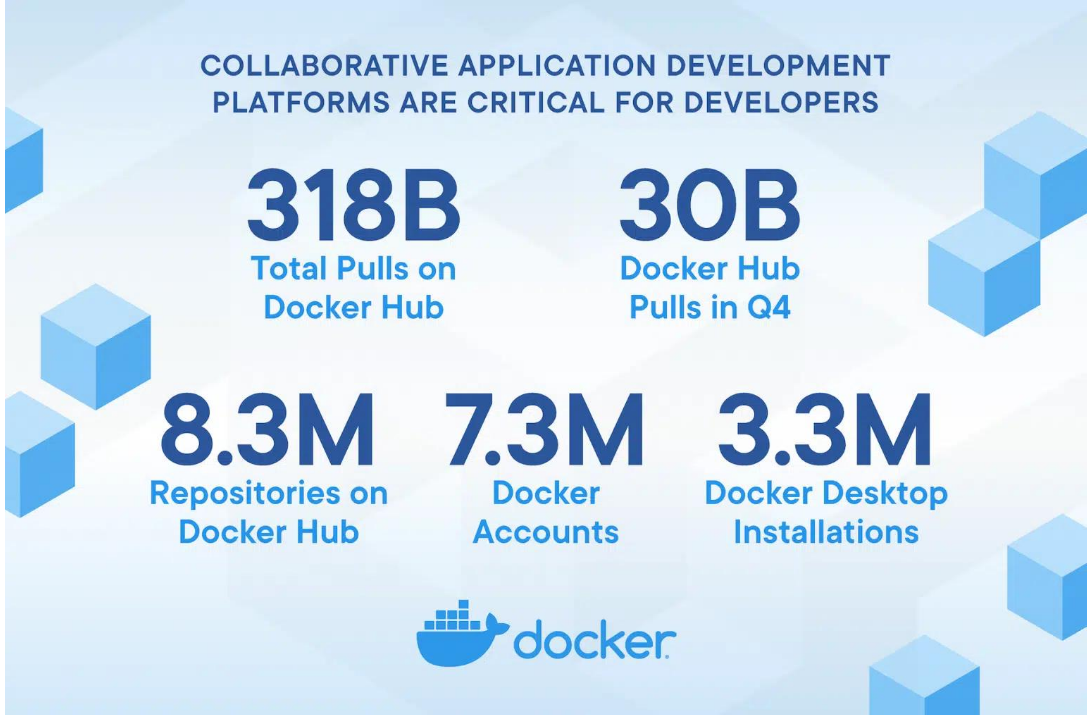
6M Repositories on Hub

5M Hub Users

2.4M Desktop Installations

*UP FROM 5.5 B A YEAR AGO

Dockerhub Feb 4 2020



COLLABORATIVE APPLICATION DEVELOPMENT PLATFORMS ARE CRITICAL FOR DEVELOPERS

318B Total Pulls on Docker Hub

30B Docker Hub Pulls in Q4

8.3M Repositories on Docker Hub

7.3M Docker Accounts

3.3M Docker Desktop Installations

docker

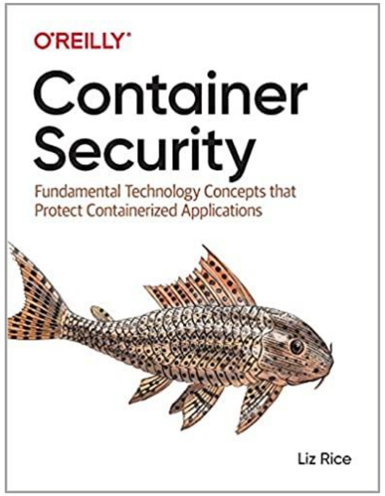
Dockerhub Feb 10 2021

- Docker image is a standard filesystem image that hosts cloud native apps
- Docker image is one of the most popular ways to deploy cloud native apps

Container Security and SCA

Along with this growth comes security risks. With millions of available images to choose from, securing containers is a dedicated discipline. There are many layers of security that apply to containers, such as:

- **The container image and software inside (image sca)**
- The configuration of image and software inside
- The interaction between the container, host operating system, and other containers on the host
- The host operating system
- Container networking and storage repositories
- The runtime environment, often in Kubernetes clusters
  - runtime auditing via log analysis



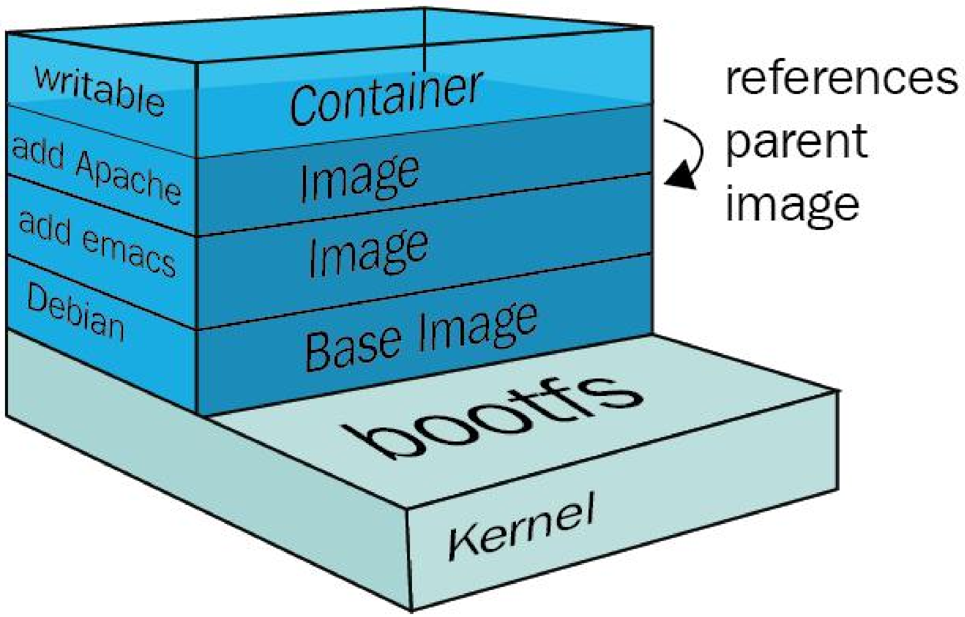
O'REILLY

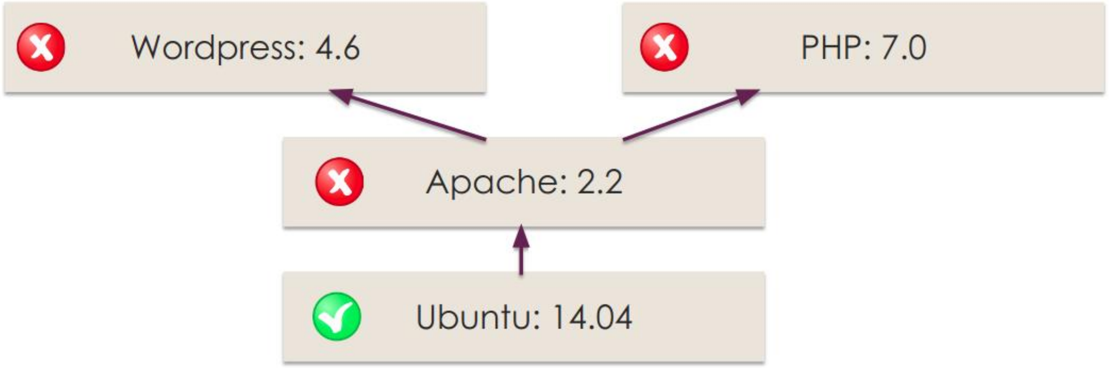
Container Security

Fundamental Technology Concepts that Protect Containerized Applications

Liz Rice

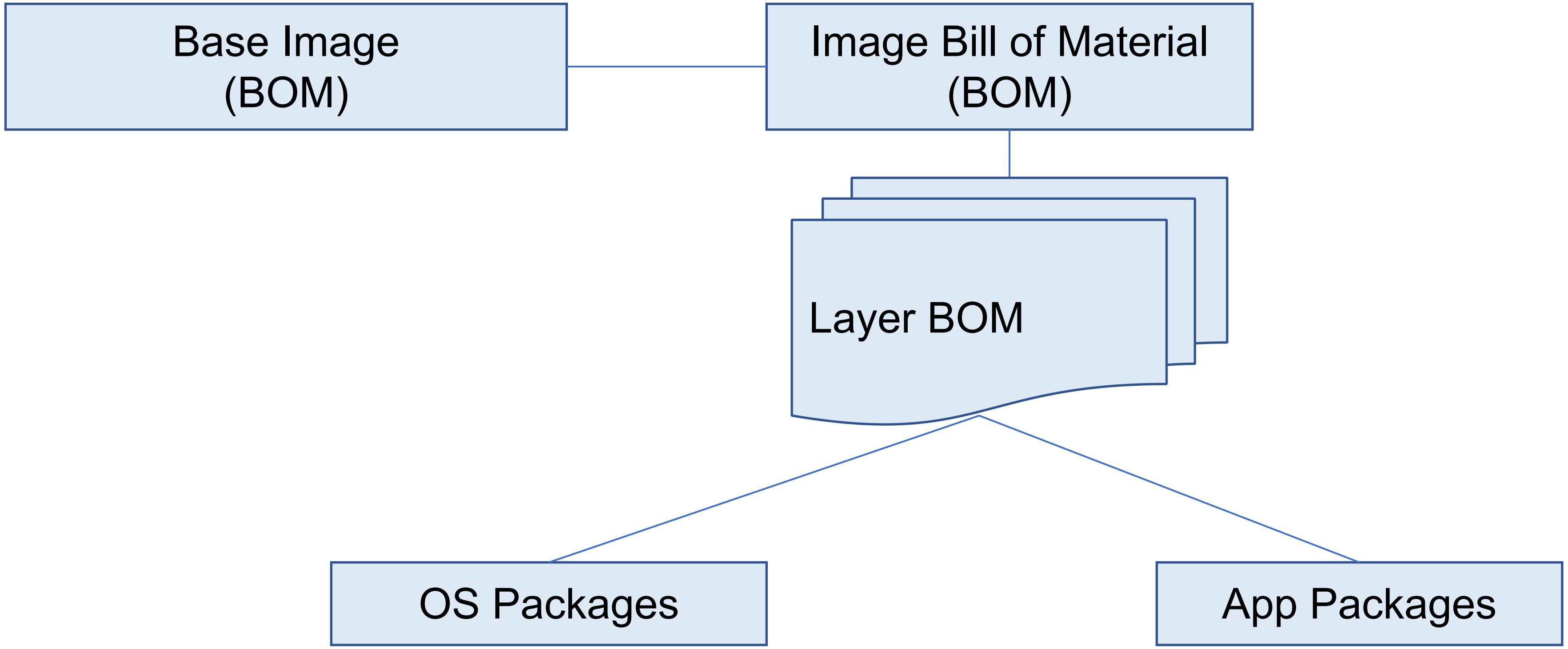
Dependencies in images



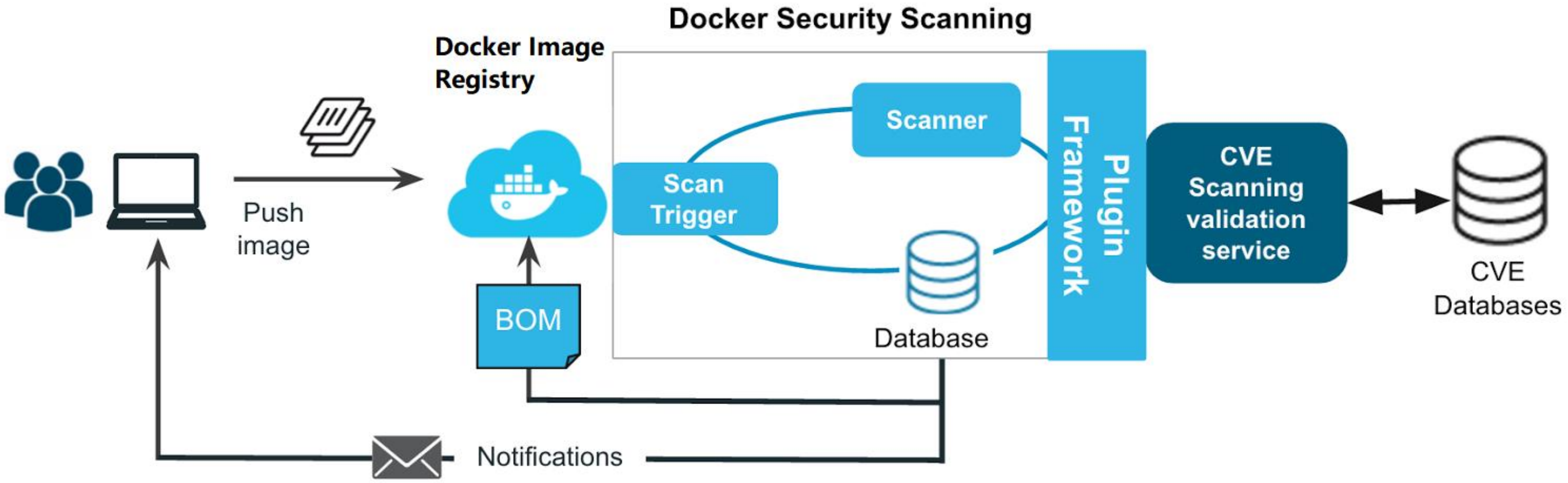


- Docker image consists of image layers
- Each layer may include one or more external dependencies or OSS libraries.

Dependencies in images



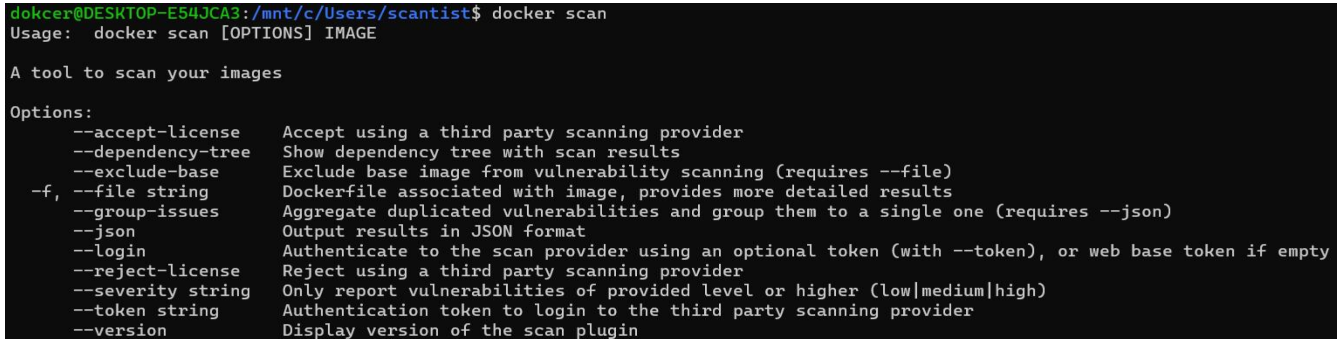
Docker Image SCA Scan



- Docker image scanning could sit alongside any docker image registry to trigger a series of events once a new image is pushed to a repository.
- The service includes a scan trigger, the scanner, a database, plugin framework and validation services that connect to CVE databases.

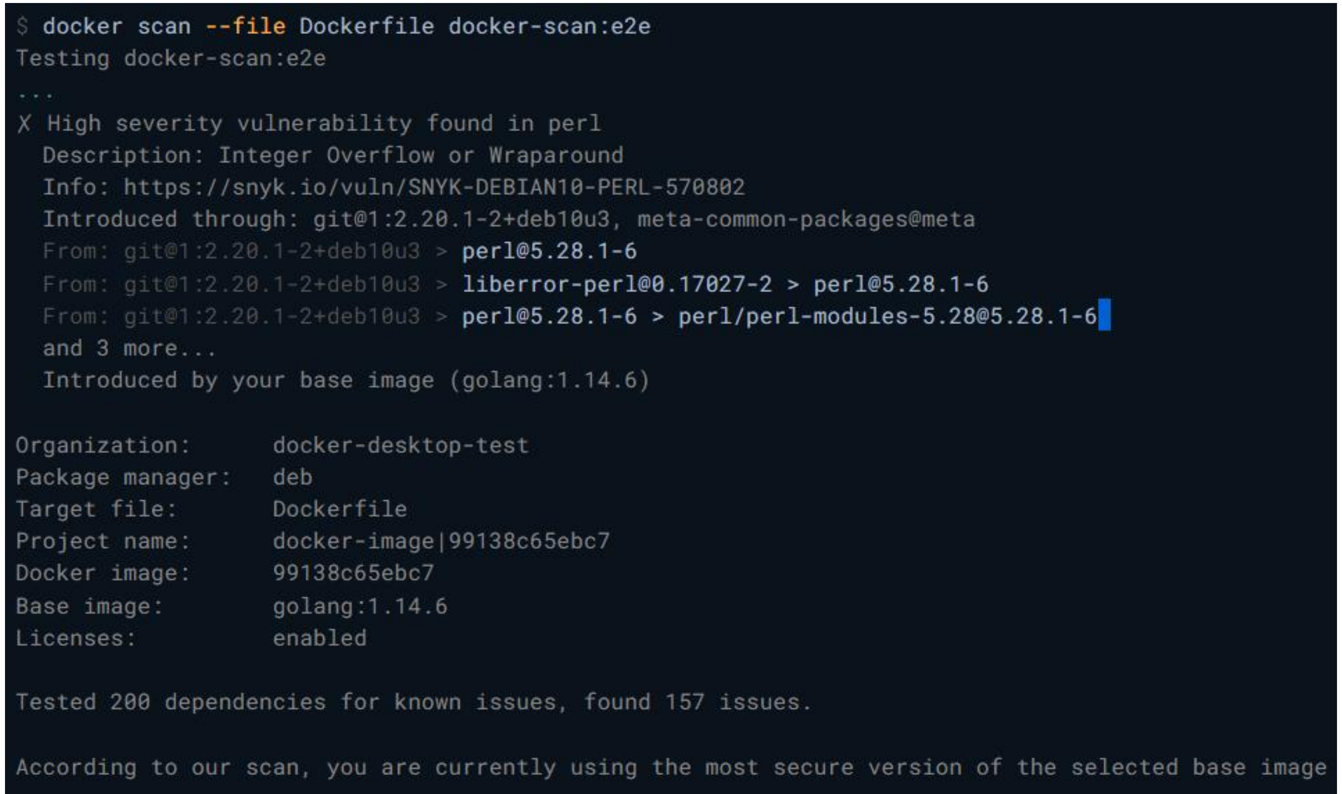
Docker Image SCA Scan

- docker scan $image



```
dokcer@DESKTOP-E54JCA3:/mnt/c/Users/scantist$ docker scan
Usage:  docker scan [OPTIONS] IMAGE

A tool to scan your images

Options:
      --accept-license    Accept using a third party scanning provider
      --dependency-tree   Show dependency tree with scan results
      --exclude-base      Exclude base image from vulnerability scanning (requires --file)
  -f, --file string       Dockerfile associated with image, provides more detailed results
      --group-issues      Aggregate duplicated vulnerabilities and group them to a single one (requires --json)
      --json              Output results in JSON format
      --login             Authenticate to the scan provider using an optional token (with --token), or web base token if empty
      --reject-license    Reject using a third party scanning provider
      --severity string   Only report vulnerabilities of provided level or higher (low|medium|high)
      --token string      Authentication token to login to the third party scanning provider
      --version           Display version of the scan plugin
```

- detect dependencies and issues

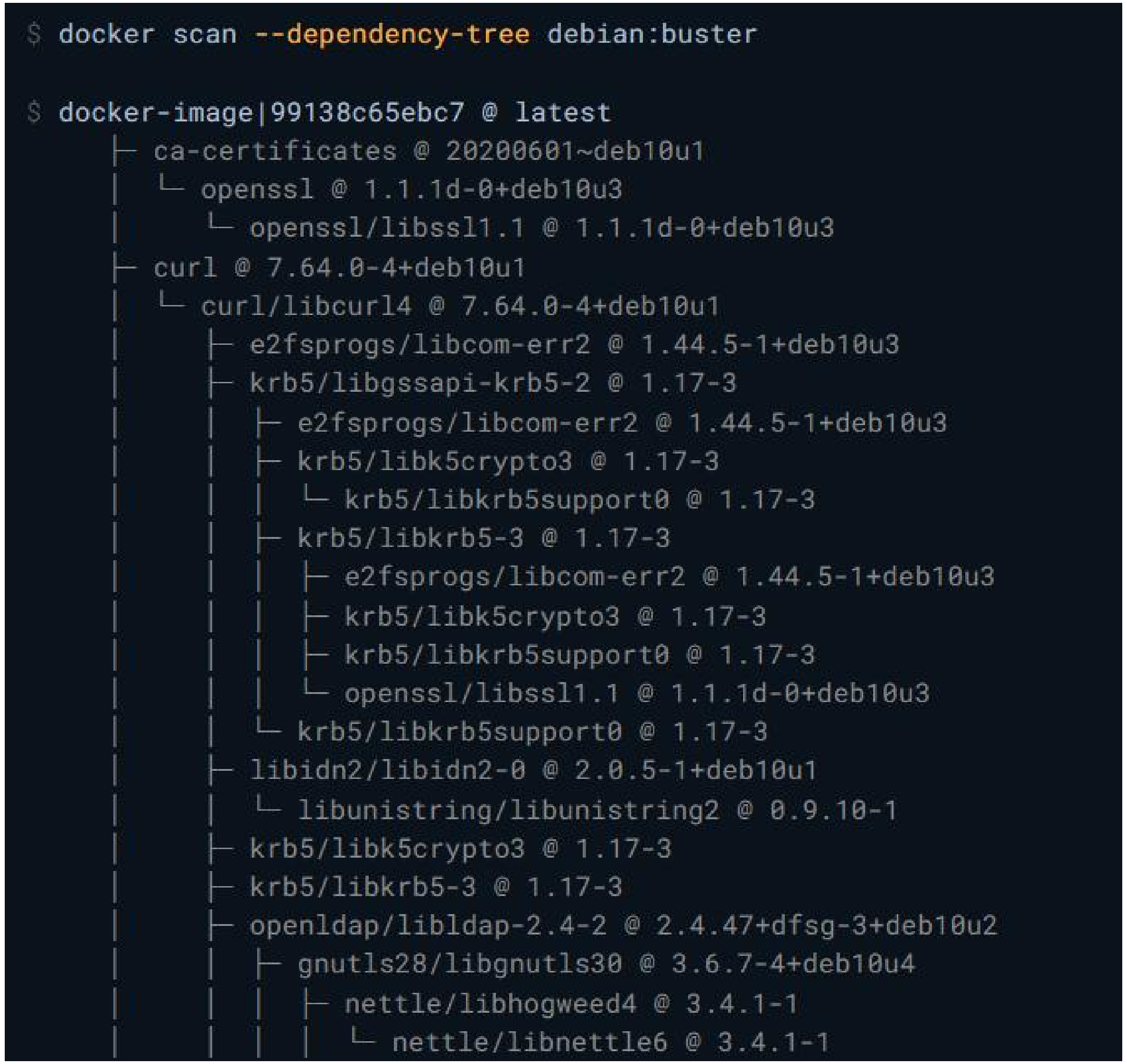


```
$ docker scan --file Dockerfile docker-scan:e2e
Testing docker-scan:e2e
...
✗ High severity vulnerability found in perl
  Description: Integer Overflow or Wraparound
  Info: https://snyk.io/vuln/SNYK-DEBIAN10-PERL-570802
  Introduced through: git@1:2.20.1-2+deb10u3, meta-common-packages@meta
  From: git@1:2.20.1-2+deb10u3 > perl@5.28.1-6
  From: git@1:2.20.1-2+deb10u3 > liberror-perl@0.17027-2 > perl@5.28.1-6
  From: git@1:2.20.1-2+deb10u3 > perl@5.28.1-6 > perl/perl-modules-5.28@5.28.1-6
  and 3 more...
  Introduced by your base image (golang:1.14.6)

Organization:      docker-desktop-test
Package manager:   deb
Target file:       Dockerfile
Project name:      docker-image|99138c65ebc7
Docker image:      99138c65ebc7
Base image:        golang:1.14.6
Licenses:          enabled

Tested 200 dependencies for known issues, found 157 issues.

According to our scan, you are currently using the most secure version of the selected base image
```

SCA for Docker

Docker Image SCA Scan

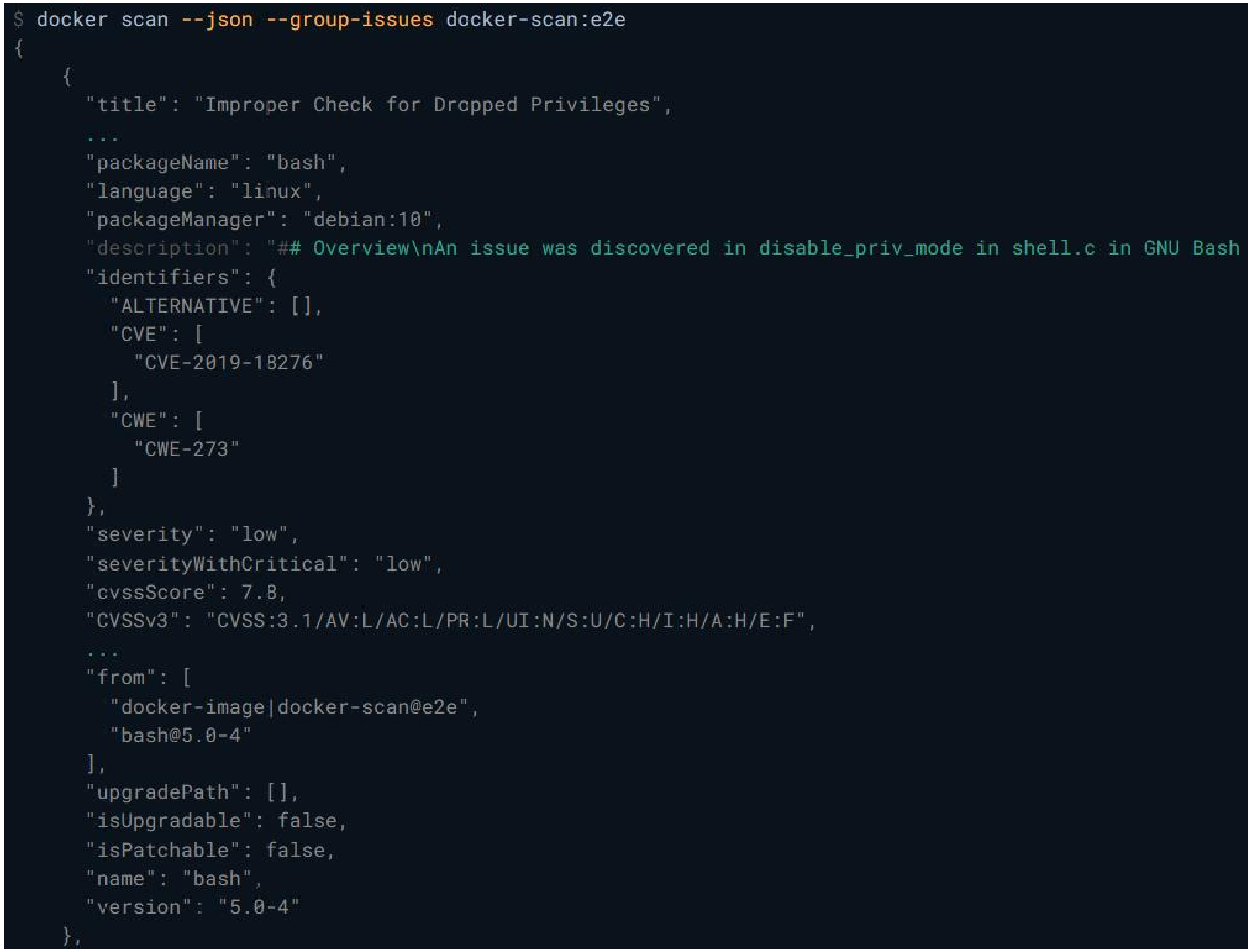
- check dependency details



```
$ docker scan --dependency-tree debian:buster

$ docker-image|99138c65ebc7 @ latest
    ├─ ca-certificates @ 20200601~deb10u1
    │  └─ openssl @ 1.1.1d-0+deb10u3
    │     └─ openssl/libssl1.1 @ 1.1.1d-0+deb10u3
    ├─ curl @ 7.64.0-4+deb10u1
    │  └─ curl/libcurl4 @ 7.64.0-4+deb10u1
    │     ├─ e2fsprogs/libcom-err2 @ 1.44.5-1+deb10u3
    │     ├─ krb5/libgssapi-krb5-2 @ 1.17-3
    │     │  ├─ e2fsprogs/libcom-err2 @ 1.44.5-1+deb10u3
    │     │  ├─ krb5/libk5crypto3 @ 1.17-3
    │     │  │  └─ krb5/libkrb5support0 @ 1.17-3
    │     │  ├─ krb5/libkrb5-3 @ 1.17-3
    │     │  │  ├─ e2fsprogs/libcom-err2 @ 1.44.5-1+deb10u3
    │     │  │  ├─ krb5/libk5crypto3 @ 1.17-3
    │     │  │  ├─ krb5/libkrb5support0 @ 1.17-3
    │     │  │  └─ openssl/libssl1.1 @ 1.1.1d-0+deb10u3
    │     │  └─ krb5/libkrb5support0 @ 1.17-3
    │     ├─ libidn2/libidn2-0 @ 2.0.5-1+deb10u1
    │     │  └─ libunistring/libunistring2 @ 0.9.10-1
    │     ├─ krb5/libk5crypto3 @ 1.17-3
    │     ├─ krb5/libkrb5-3 @ 1.17-3
    │     ├─ openldap/libldap-2.4-2 @ 2.4.47+dfsg-3+deb10u2
    │     │  ├─ gnutls28/libgnutls30 @ 3.6.7-4+deb10u4
    │     │  │  ├─ nettle/libhogweed4 @ 3.4.1-1
    │     │  │  │  └─ nettle/libnettle6 @ 3.4.1-1
```

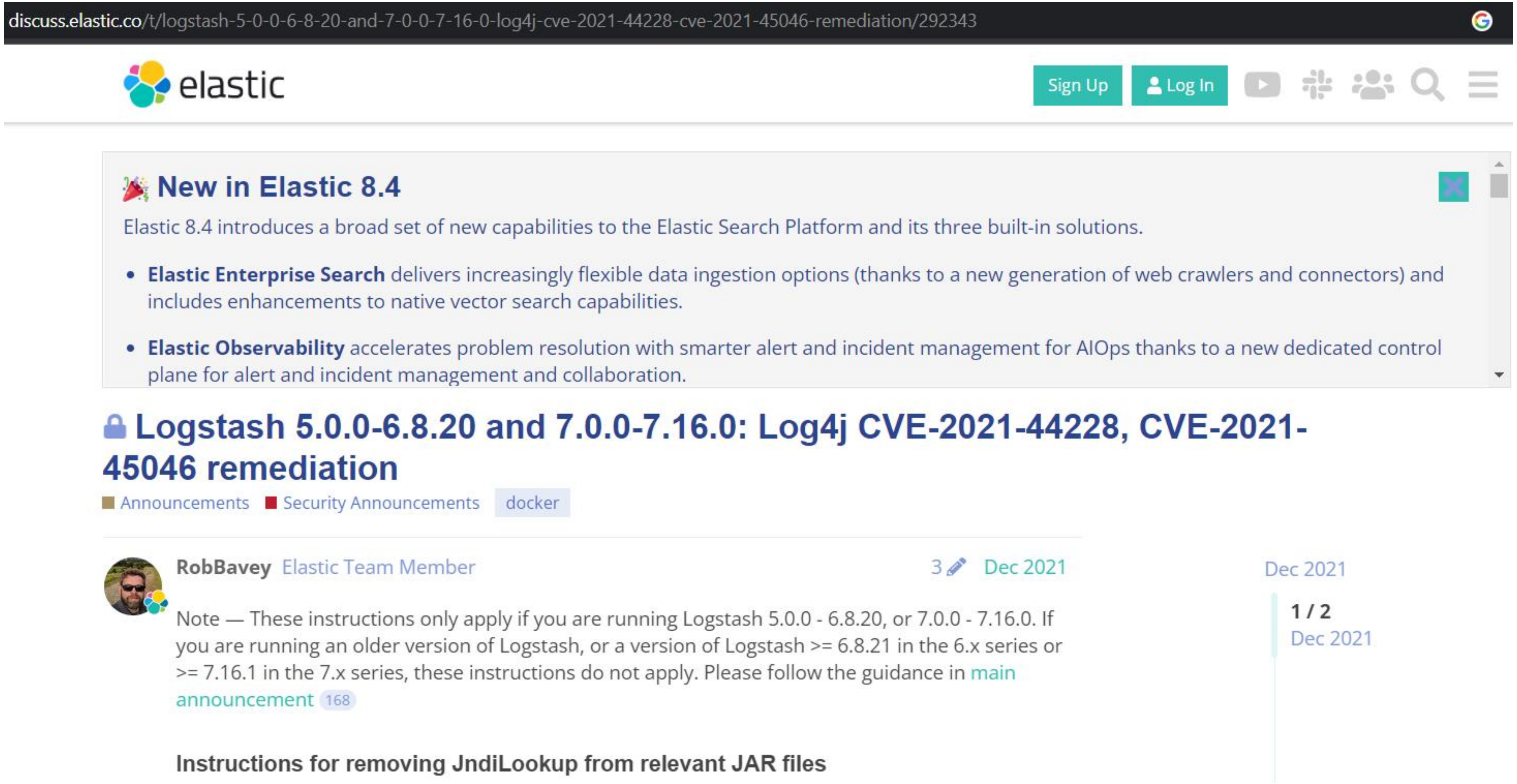
Docker Image SCA Scan

- check issue details



```
$ docker scan --json --group-issues docker-scan:e2e
{
    {
      "title": "Improper Check for Dropped Privileges",
      ...
      "packageName": "bash",
      "language": "linux",
      "packageManager": "debian:10",
      "description": "## Overview\nAn issue was discovered in disable_priv_mode in shell.c in GNU Bash
      "identifiers": {
        "ALTERNATIVE": [],
        "CVE": [
          "CVE-2019-18276"
        ],
        "CWE": [
          "CWE-273"
        ]
      },
      "severity": "low",
      "severityWithCritical": "low",
      "cvssScore": 7.8,
      "CVSSv3": "CVSS:3.1/AV:L/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H/E:F",
      ...
      "from": [
        "docker-image|docker-scan@e2e",
        "bash@5.0-4"
      ],
      "upgradePath": [],
      "isUpgradable": false,
      "isPatchable": false,
      "name": "bash",
      "version": "5.0-4"
    },
```

Example: detect log4j (CVE-2021-44228)



discuss.elastic.co/t/logstash-5-0-0-6-8-20-and-7-0-0-7-16-0-log4j-cve-2021-44228-cve-2021-45046-remediation/292343

New in Elastic 8.4

Elastic 8.4 introduces a broad set of new capabilities to the Elastic Search Platform and its three built-in solutions.

- **Elastic Enterprise Search** delivers increasingly flexible data ingestion options (thanks to a new generation of web crawlers and connectors) and includes enhancements to native vector search capabilities.
- **Elastic Observability** accelerates problem resolution with smarter alert and incident management for AIOps thanks to a new dedicated control plane for alert and incident management and collaboration.

Logstash 5.0.0-6.8.20 and 7.0.0-7.16.0: Log4j CVE-2021-44228, CVE-2021-45046 remediation

Announcements · Security Announcements · docker

RobBavey Elastic Team Member 3 Dec 2021

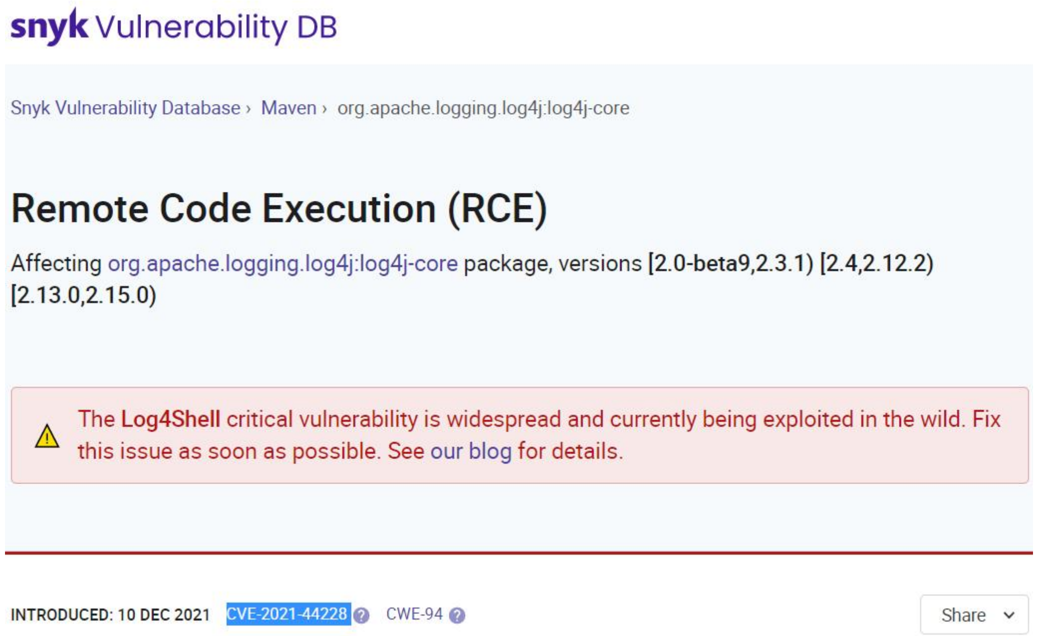
Note — These instructions only apply if you are running Logstash 5.0.0 - 6.8.20, or 7.0.0 - 7.16.0. If you are running an older version of Logstash, or a version of Logstash >= 6.8.21 in the 6.x series or >= 7.16.1 in the 7.x series, these instructions do not apply. Please follow the guidance in main announcement 168

Instructions for removing JndiLookup from relevant JAR files

Example: detect log4j (CVE-2021-44228)

- docker pull docker.elastic.co/logstash/logstash:7.3.1
- docker scan docker.elastic.co/logstash/logstash:7.3.1 --dependency-tree
  - Analyzing container dependencies for docker.elastic.co/logstash/logstash:7.3.1
  - Querying vulnerabilities database…

```
/*************  logstash731_dockerscan.log (line 11509)
…..
   introduced by org.apache.logging.log4j:log4j-core@2.11.1
  ✗ Remote Code Execution (RCE) [Critical Severity][https://security.snyk.io/vuln/SNYK-JAVA-ORGAPACHELOGGINGLOG4J-
2314720] in org.apache.logging.log4j:log4j-core@2.11.1
…

*****************/
```



snyk Vulnerability DB

Snyk Vulnerability Database › Maven › org.apache.logging.log4j:log4j-core

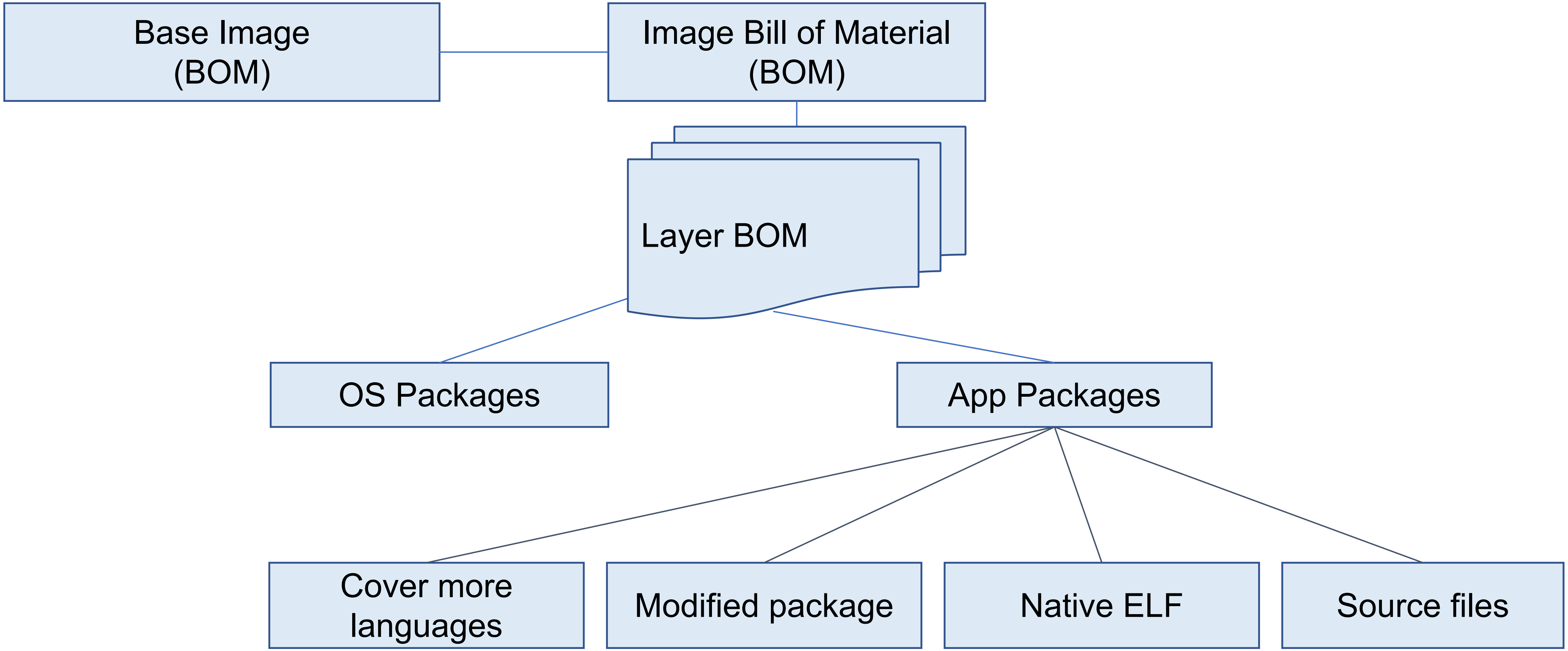
Remote Code Execution (RCE)

Affecting org.apache.logging.log4j:log4j-core package, versions [2.0-beta9,2.3.1) [2.4,2.12.2) [2.13.0,2.15.0)

⚠ The Log4Shell critical vulnerability is widespread and currently being exploited in the wild. Fix this issue as soon as possible. See our blog for details.

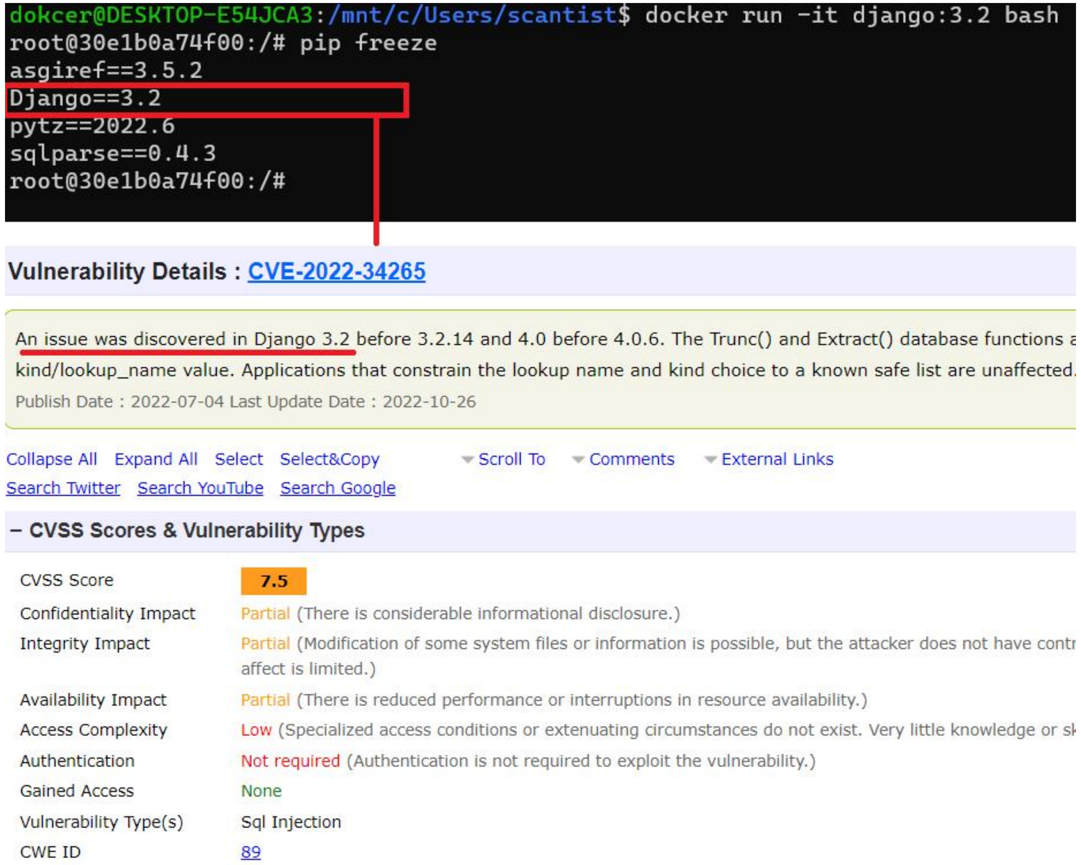
INTRODUCED: 10 DEC 2021 CVE-2021-44228 CWE-94

Complex situations and challenges



Complex situations and challenges

- Cover more languages and package managers in docker scan
  - https://scantist.atlassian.net/wiki/spaces/SD/pages/302841894/Supported+Languages+and+Formats
- Example
  - Django 3.2
  - docker scan django:3.2 |grep django
    - cannot detect this component
    - django32_dockerscan.log



```
dokcer@DESKTOP-E54JCA3:/mnt/c/Users/scantist$ docker run -it django:3.2 bash
root@30e1b0a74f00:/# pip freeze
asgiref==3.5.2
Django==3.2
pytz==2022.6
sqlparse==0.4.3
root@30e1b0a74f00:/#
```

Vulnerability Details : CVE-2022-34265

An issue was discovered in Django 3.2 before 3.2.14 and 4.0 before 4.0.6. The Trunc() and Extract() database functions a
kind/lookup_name value. Applications that constrain the lookup name and kind choice to a known safe list are unaffected.

Publish Date : 2022-07-04 Last Update Date : 2022-10-26

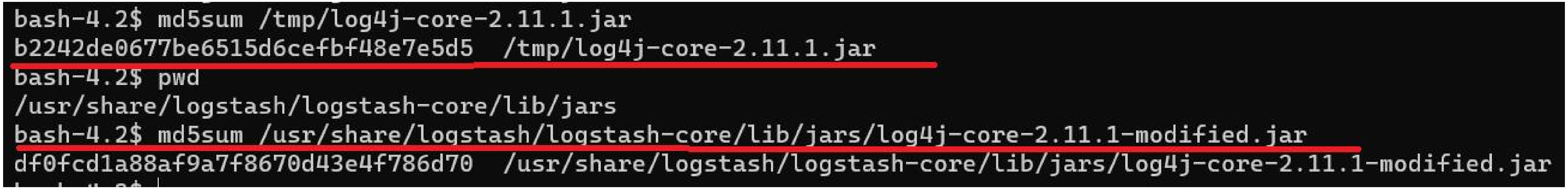
Collapse All Expand All Select Select&Copy ▾ Scroll To ▾ Comments ▾ External Links
Search Twitter Search YouTube Search Google

– CVSS Scores & Vulnerability Types

CVSS Score	7.5
Confidentiality Impact	Partial (There is considerable informational disclosure.)
Integrity Impact	Partial (Modification of some system files or information is possible, but the attacker does not have contr affect is limited.)
Availability Impact	Partial (There is reduced performance or interruptions in resource availability.)
Access Complexity	Low (Specialized access conditions or extenuating circumstances do not exist. Very little knowledge or sk
Authentication	Not required (Authentication is not required to exploit the vulnerability.)
Gained Access	None
Vulnerability Type(s)	Sql Injection
CWE ID	89

Complex situations and challenges

- **Modified packages, due to:**
  - inhouse compiled
  - vendor modified
  - maliciously hacked
- Example
  - we modified the log4j-core.jar in docker.elastic.co/logstash/logstash:7.3.1
    - by removing the pom.xml inside
    - then it cannot be detected anymore
    - logstash_modified_dockerscan.log



```
bash-4.2$ md5sum /tmp/log4j-core-2.11.1.jar
b2242de0677be6515d6cefbf48e7e5d5  /tmp/log4j-core-2.11.1.jar
bash-4.2$ pwd
/usr/share/logstash/logstash-core/lib/jars
bash-4.2$ md5sum /usr/share/logstash/logstash-core/lib/jars/log4j-core-2.11.1-modified.jar
df0fcd1a88af9a7f8670d43e4f786d70  /usr/share/logstash/logstash-core/lib/jars/log4j-core-2.11.1-modified.jar
bash-4.2$
```

Complex situations and challenges

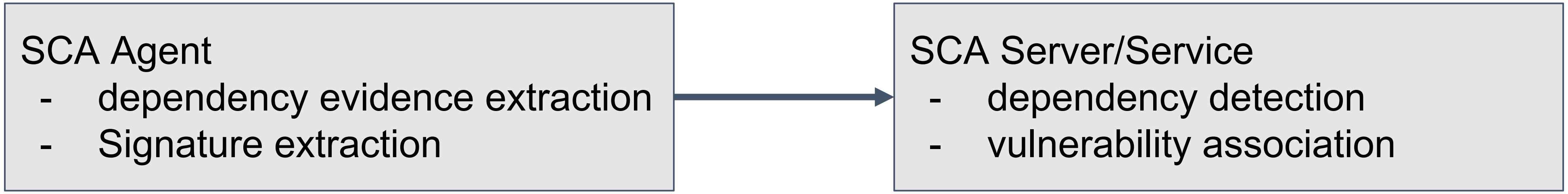
- **Native ELF, installed by:**
  - wget, curl… (web download)
  - docker cp
  - compiled in the image directly (c libs, so files)
- Example
  - ffmepg 3.4.2 with **CVE-2018-7557**
  - docker pull ubuntu:18.04
  - cid=$(docker run -dt docker.io/library/ubuntu:18.04)
  - docker cp ffmepg.3.4.2.so $cid:/tmp
  - docker commit $cid modified_img
  - docker scan modified_img
  - ffmepg342.log   (cannot detect the ffmpeg ELF file)

Complex situations and challenges

- **Source files, installed by:**
  - wget, curl… (web download)
  - docker cp
  - compiled in the image directly (inhouse compiled package, e.g: openssl or npm package)
- Example
  - openssl 1.0.2g
  - npm lodash dist

Complex situations and challenges

1. support more languages and package managers
2. provide signature based match
    a. current matching logic is mainly name based or hash based
        i. pom.xml
        ii. file hash
    b. Signature based: AST structure and code signatures
        i. https://scantist.io/u/xyz031702/org/xyz031702/projects/10664/scans/102438/library?tab=0
        ii. Automated third-party library detection for android applications: Are we there yet? X Zhan, L Fan, T Liu, S Chen, L Li, H Wang, Y Xu… - 2020 35th IEEE/ACM International
    c. data protection and transfer for signatures



SCA Agent
- dependency evidence extraction
- Signature extraction

SCA Server/Service
- dependency detection
- vulnerability association

Best practices for developing docker image

Building secure images is a continuous process. Consider the recommendations and best practices highlighted in this guide to plan and build efficient, scalable, and secure images.

- Start with a base image that you trust. Remember the Official image and Verified Publisher badges when you choose your base images.
- Secure your code and its dependencies.
- Select a minimal base image which contains only the required packages.
- Use multi-stage builds to optimize your image.
- Ensure you carefully monitor and manage the tools and dependencies you add to your image.
- Ensure you scan images at multiple stages during your development lifecycle.
- Check your images frequently for vulnerabilities.

https://docs.docker.com/develop/scan-images/

Overview

SCA for Docker

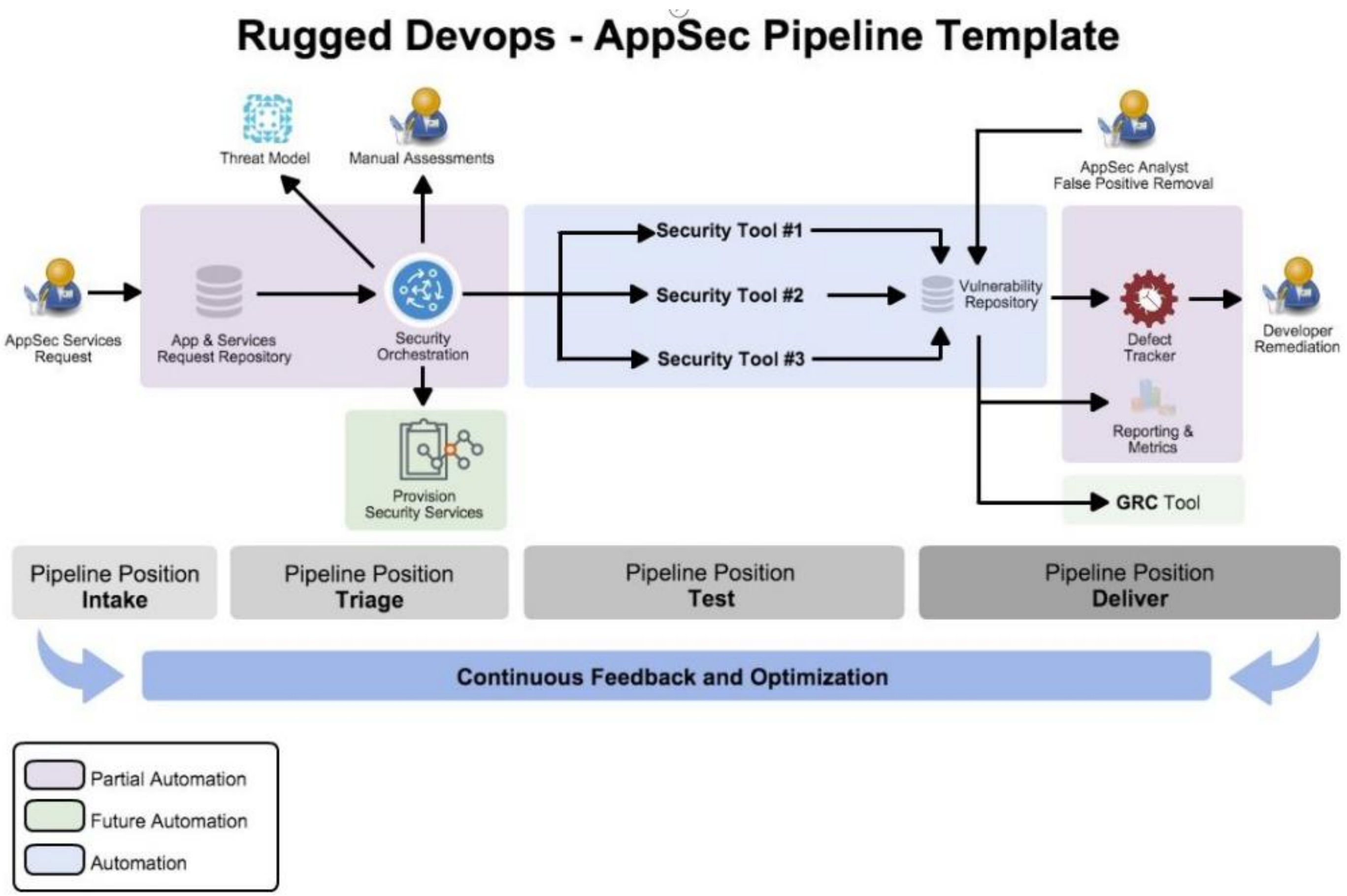
- SCA scan for docker images
- challenges
- effort to improve

Docker for SCA

- docker as a sandbox for SCA
- challenges
- effort to improve

Disclaimer: all the figures from internet is for sharing and education usage only

AppSec Pipeline and Orchestration



DevSecOps pipeline focuses more on automation.

- Security = tool scanning
- Automate security

(Engineer’s system aspect)

ASTO: AppSec Testing Orchestration

ASTO highlights that manual auditing is not avoidable. **(Security experter’s system aspect)**

ASTO focuses to easy the following work:

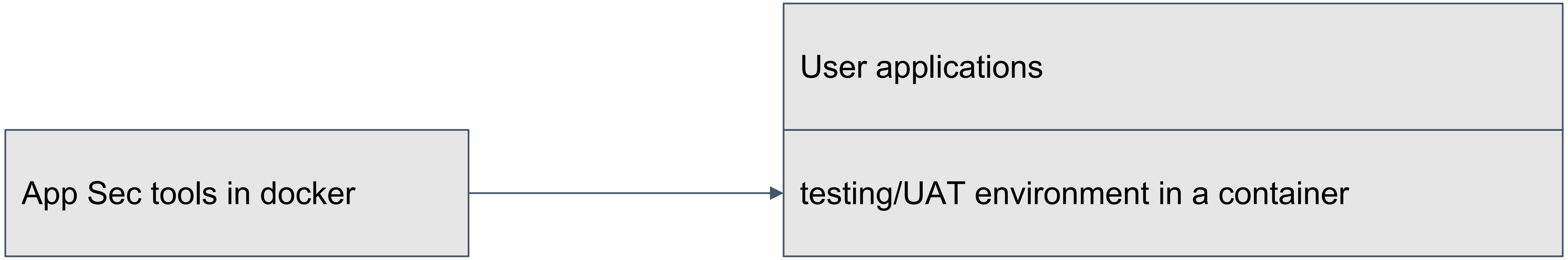
- Result aggregations from scanners
- Centralized board for auditing work
- Task tracking
  - Create issues
  - Track issues
  - Assignees, stakeholders

AppSec Pipeline and Orchestration

In reality

- Security team wants to trigger AppSec tools (e.g.: SCA), but
    - no access the CI/CD pipelines
    - no interfere with normal development workflow
    - no clue with the environment dependencies
- surprisingly, in quite some companies, there might be many applications (source code repos > 10k), but there are limited environments (<100)
- strong associations between application and environments: java applications ←→ JDK11 + Maven

Docker can help !

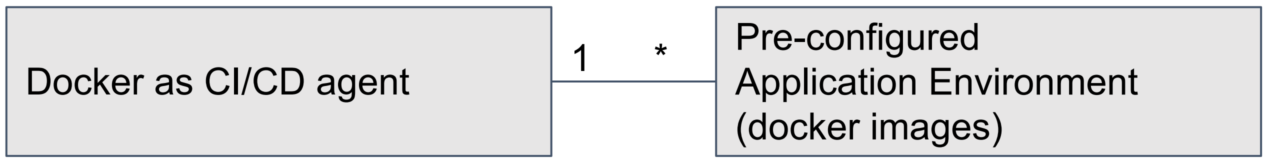


App Sec tools in docker

User applications

testing/UAT environment in a container

Docker for SCA



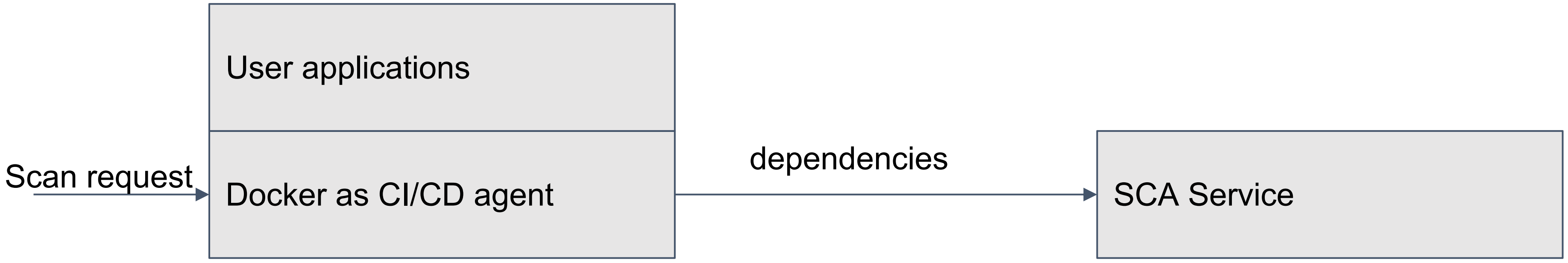
Docker as CI/CD agent

1

*

Pre-configured Application Environment (docker images)

- **Post build SCA Scan**



User applications

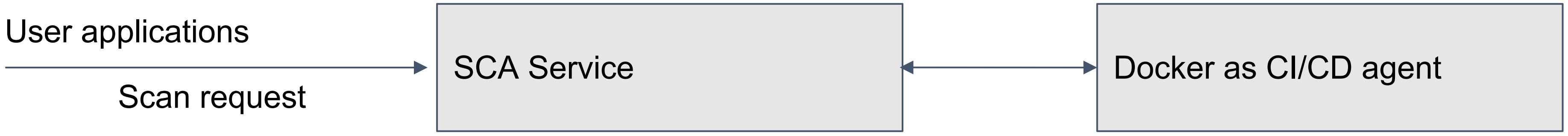
Scan request

Docker as CI/CD agent

dependencies

SCA Service

- **Pre build SCA Scan**



User applications

Scan request

SCA Service

Docker as CI/CD agent

ABOUT US

Scantist was founded in 2016 as a spin—off from the world-leading Cyber Security Lab at Nanyang Technological University, Singapore.

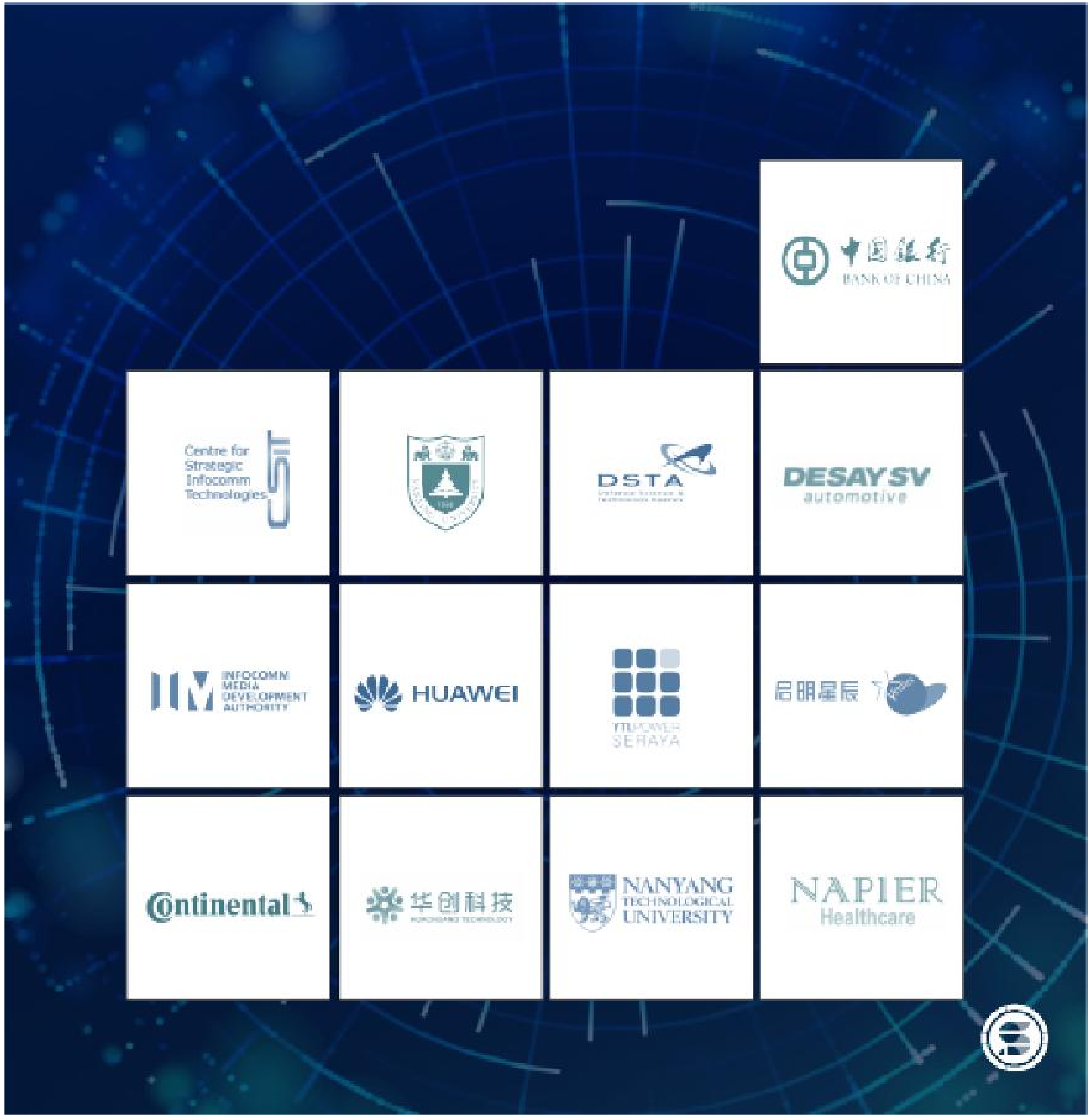
We are the recipient of the 2020 CSA innovation Award as well as 2018 NRF National Cybersecurity Research Grant.

We currently employ a 51-member strong team across our three offices in Singapore, Mumbai and Shanghai.

18 ENTERPRISE

07 SME

06 GOVERNMENT



NTU-Scantist DevSecOps Professional & Tools course

(Synchronous & Asynchronous e-Learning)

3 certificates in 1 course

from

NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE DevOps INSTITUTE SCANTIST

Who should attend?

- Cybersecurity Professionals & Consultants
- Developers
- Risk and compliance managers

Course outline

- DevOps Institute model
- DevSecOps Tool chain
- CVE Triage & Vulnerability management

Course Availability

- Date(s): 23 to 28 Nov 2022
- Time: 9:00AM to 5:00PM (Day 1: Briefing and Self-learning, Day 2-4: Facilitated learning)
- Venue: Virtual (Online) & NTU e-Learning Platform
- Registration Closing Date: 10 Nov 2022

Cohort 1 2022 Class Participants

DBS

EVANTEK
DRIVING INNOVATION

TERRA
SYSTEMS

CYSECURE

InsiderSecurity

CloudNet

TELOGIC

adexel

Quadrant360 Consulting
Security Consulting and
Managed Security Services

Cohort 2 ; Oct 2022 Class Participants

ncs

cisco

INSYGHTS SECURITY

ST Engineering

Rio Tinto

COMPUTER GUYS

GoalsMapper

Cohort 2 ; Oct 2022 Class Participants

1eBS
Automate • Engage • Transform

NETPLUZ
good communications simplified

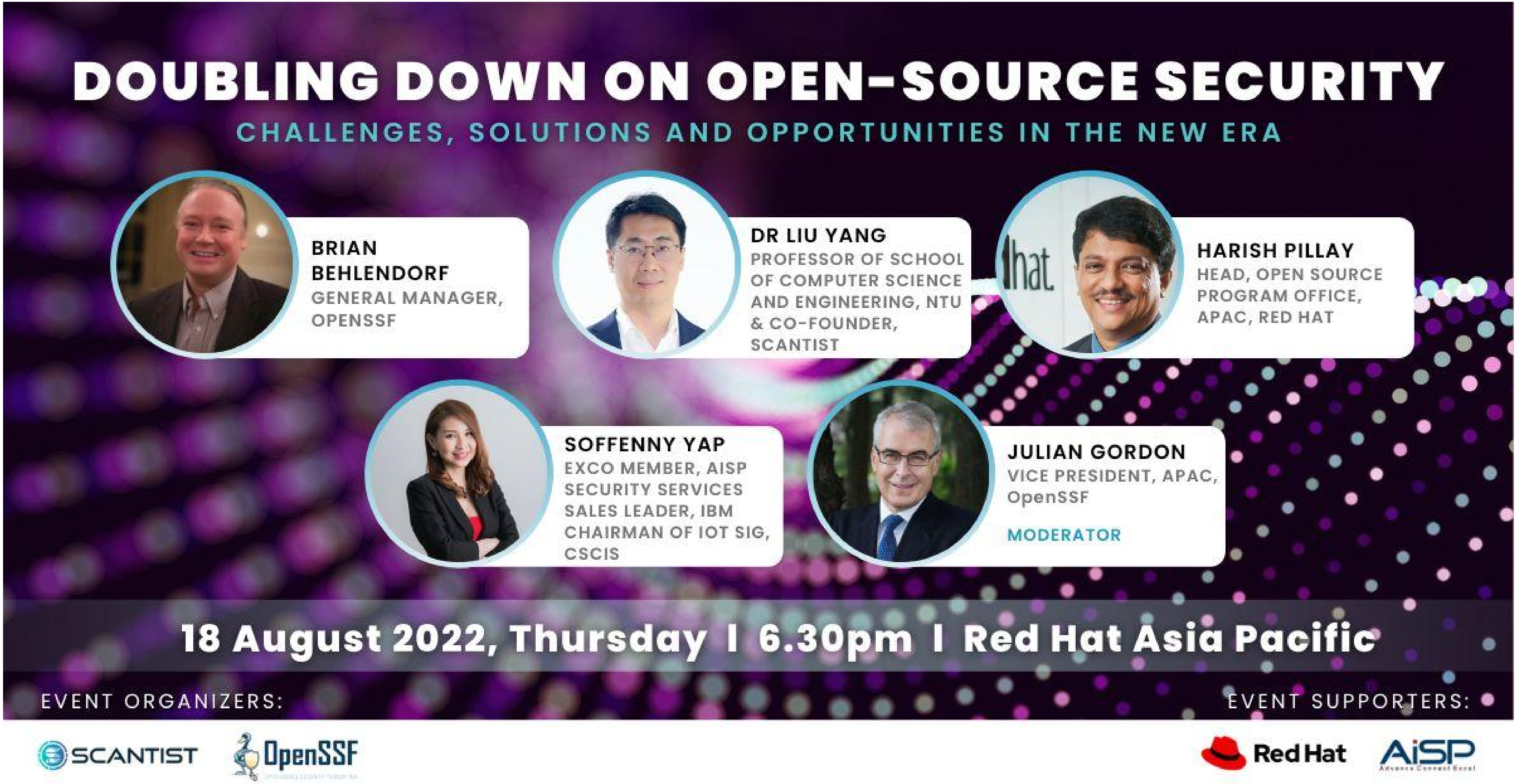
DBS

BEYONDSOFT®
beyond your expectations

NUMEN

SCANTIST

Scantist DevSecOps Professional and Tools Certification



DOUBLING DOWN ON OPEN-SOURCE SECURITY

CHALLENGES, SOLUTIONS AND OPPORTUNITIES IN THE NEW ERA

BRIAN BEHLENDORF
GENERAL MANAGER, OPENSSF

DR LIU YANG
PROFESSOR OF SCHOOL OF COMPUTER SCIENCE AND ENGINEERING, NTU & CO-FOUNDER, SCANTIST

HARISH PILLAY
HEAD, OPEN SOURCE PROGRAM OFFICE, APAC, RED HAT

SOFFENNY YAP
EXCO MEMBER, AISP
SECURITY SERVICES SALES LEADER, IBM
CHAIRMAN OF IOT SIG, CSCIS

JULIAN GORDON
VICE PRESIDENT, APAC, OpenSSF

MODERATOR

18 August 2022, Thursday I 6.30pm I Red Hat Asia Pacific

EVENT ORGANIZERS:

SCANTIST OpenSSF

EVENT SUPPORTERS:

Red Hat AiSP

OpenSSF x Scantist x Red Hat x AiSP

DOUBLING DOWN ON OPEN SOURCE SECURITY

18 August 2022

JULIAN GORDON OpenSSF
DOUBLING DOWN ON OPEN SOURCE SECURITY
SOFFENNY YAP AiSP
R LIU YANG Scantist
RIAN ENDORF enSSF
SSID: Red Hat Guest
PW: Shadowman35
Scantick Lite - Open Source Security (OSS) Report
SIGN UP FOR A FREE OSS HEALTH CHECK NOW!

THERE IS NO TGIF FOR OPEN SOURCE SECURITY

BRIAN BEHLENDORF
GENERAL MANAGER, OPENSSF

DR LIU YANG
PROFESSOR, NTU & CO-FOUNDER, SCANTIST

LLEWELLAN VANCE
SPARK STARTUP ECOSYSTEM LEAD, HUAWEI

ASANKHAYA SHARMA
DIRECTOR OF INCUBATION, VERACODE

ROHAN SOOD
COO, SCANTIST

MODERATOR

AUGUST 19 2022, FRIDAY
5.30PM @ ICE71

EVENT ORGANIZERS:

SCANTIST
OpenSSF
OPEN SOURCE SECURITY FOUNDATION

EVENT SUPPORTERS:

ICE71
HUAWEI
SPARK Accelerator

OpenSSF x Scantist x Ice71 x Huawei Cloud

THERE IS NO TGIF FOR OPEN SOURCE SECURITY

19 August 2022

SCANTIST

Thank you!