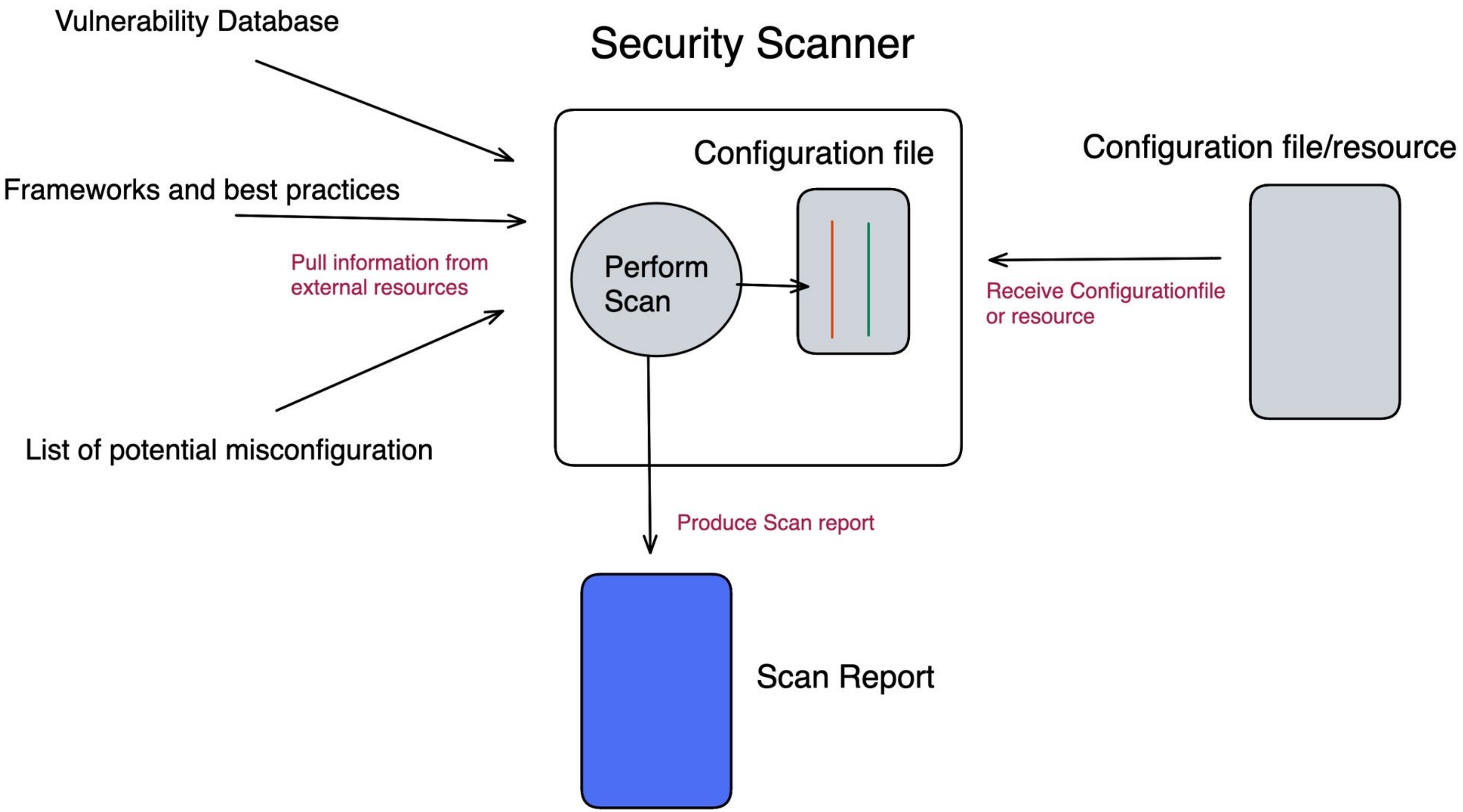
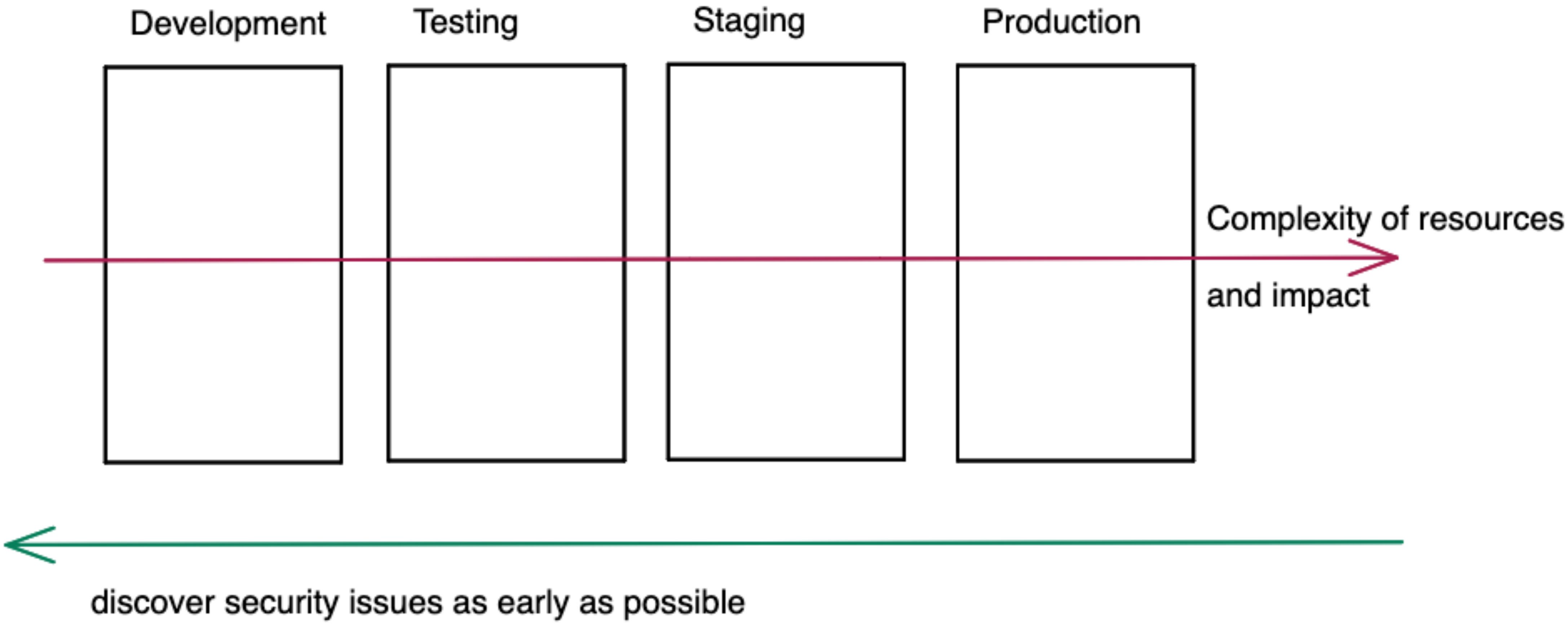
Aqua Security Trivy Overview

January 2023

What is security scanning

How do Security Scanners Work?


Vulnerability Database
Frameworks and best practices
Pull information from external resources
List of potential misconfiguration
Security Scanner
Configuration file
Perform Scan
Configuration file/resource
Receive Configurationfile or resource
Produce Scan report
Scan Report



Development

Testing

Staging

Production

Complexity of resources and impact

discover security issues as early as possible

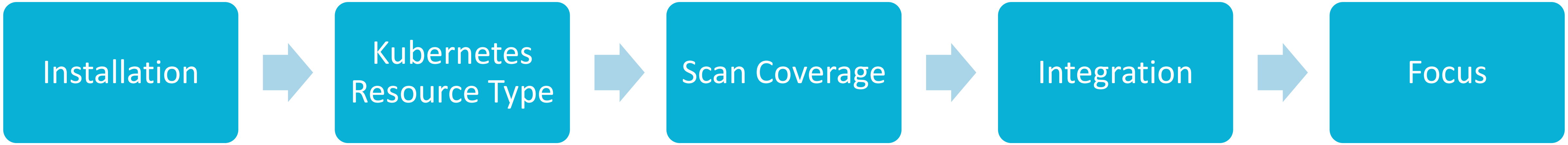
Trivy - https://github.com/aquasecurity/trivy

- All in one, cloud native security scanner
- Open source, 20.000 stars on GitHub
- Used as security scanner in Aqua Enterprise
- The Trivy Operator – Kubernetes native Operator
- Adopted across the cloud native ecosystem by companies such as Wise Engineering, Microsoft, Lyft, Mergify and many more
- 25+ Integrations with other projects in the cloud native ecosystem

⭐ Give us a Star on GitHub ⭐

"Trivy was a clear leader in the market as far as features, functionality, and capabilities," said Sam White, Sr. Product Manager at GitLab

Difference between Security Scanners



Installation

Kubernetes Resource Type

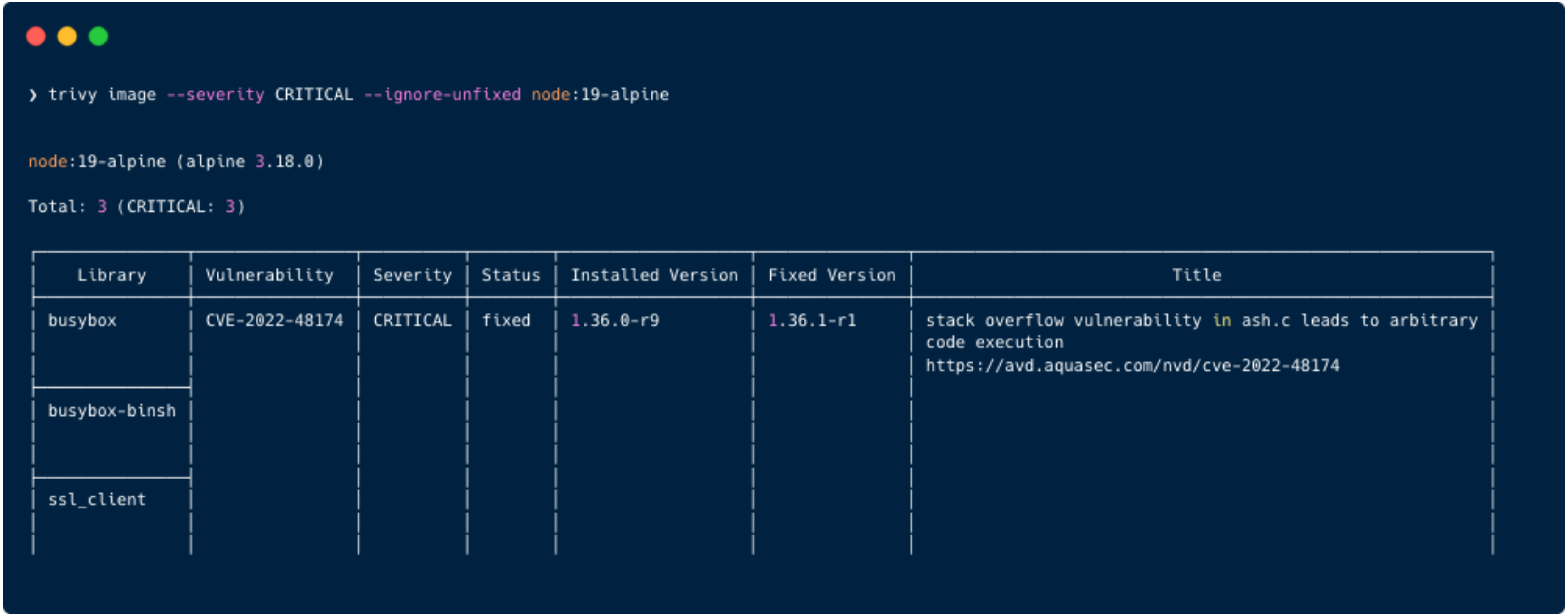
Scan Coverage

Integration

Focus

Trivy is divided into 4 main scanners

Vulnerabilities



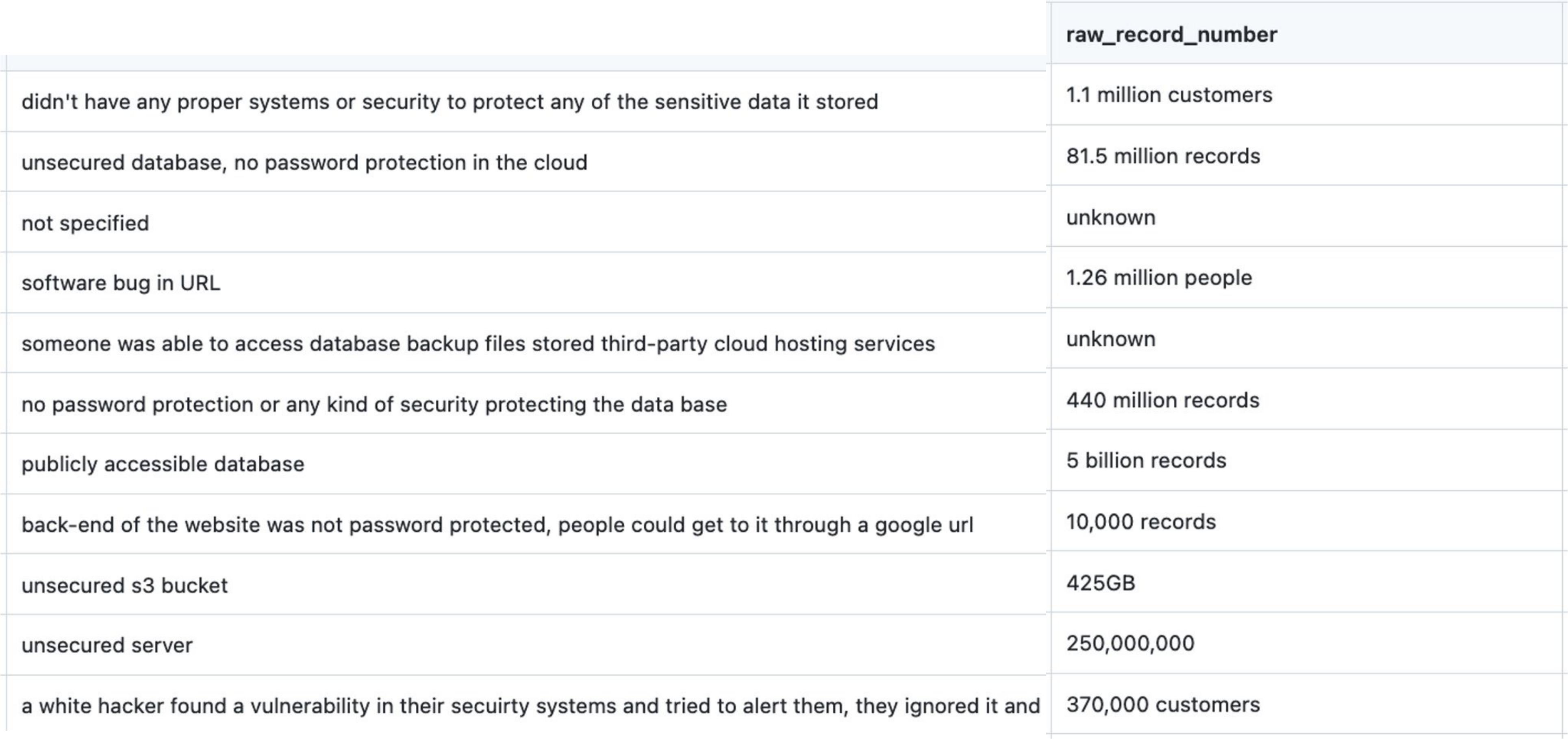
```
❯ trivy image --severity CRITICAL --ignore-unfixed node:19-alpine


node:19-alpine (alpine 3.18.0)

Total: 3 (CRITICAL: 3)

┌───────────────┬────────────────┬──────────┬────────┬───────────────────┬───────────────┬──────────────────────────────────────────────────────────┐
│    Library    │ Vulnerability  │ Severity │ Status │ Installed Version │ Fixed Version │                          Title                           │
├───────────────┼────────────────┼──────────┼────────┼───────────────────┼───────────────┼──────────────────────────────────────────────────────────┤
│ busybox       │ CVE-2022-48174 │ CRITICAL │ fixed  │ 1.36.0-r9         │ 1.36.1-r1     │ stack overflow vulnerability in ash.c leads to arbitrary │
│               │                │          │        │                   │               │ code execution                                           │
│               │                │          │        │                   │               │ https://avd.aquasec.com/nvd/cve-2022-48174               │
├───────────────┤                │          │        │                   │               │                                                          │
│ busybox-binsh │                │          │        │                   │               │                                                          │
│               │                │          │        │                   │               │                                                          │
│               │                │          │        │                   │               │                                                          │
├───────────────┤                │          │        │                   │               │                                                          │
│ ssl_client    │                │          │        │                   │               │                                                          │
│               │                │          │        │                   │               │                                                          │
│               │                │          │        │                   │               │                                                          │
```

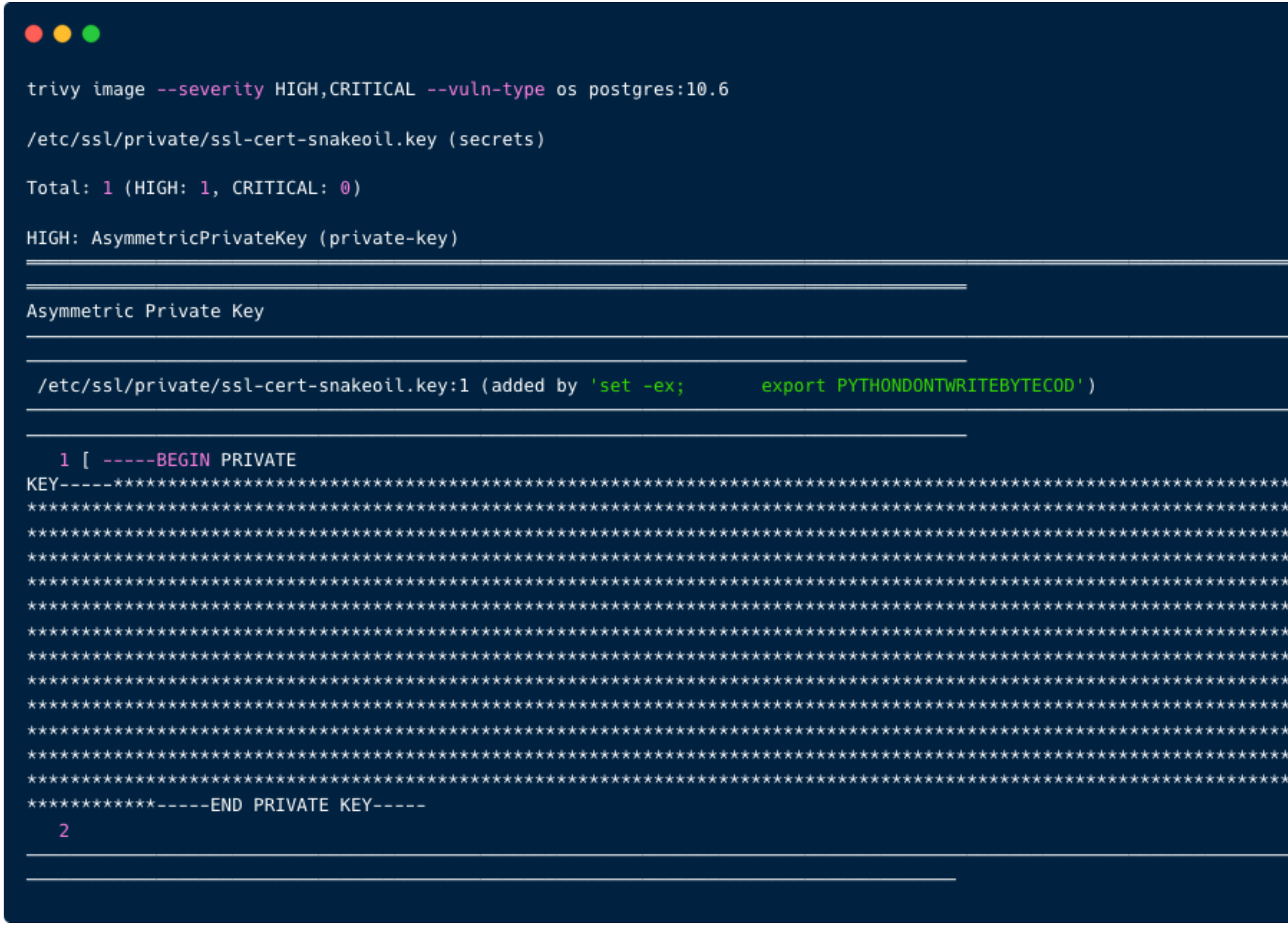
Misconfiguration



	raw_record_number
didn't have any proper systems or security to protect any of the sensitive data it stored	1.1 million customers
unsecured database, no password protection in the cloud	81.5 million records
not specified	unknown
software bug in URL	1.26 million people
someone was able to access database backup files stored third-party cloud hosting services	unknown
no password protection or any kind of security protecting the data base	440 million records
publicly accessible database	5 billion records
back-end of the website was not password protected, people could get to it through a google url	10,000 records
unsecured s3 bucket	425GB
unsecured server	250,000,000
a white hacker found a vulnerability in their secuirty systems and tried to alert them, they ignored it and	370,000 customers

Source

Exposed Secrets



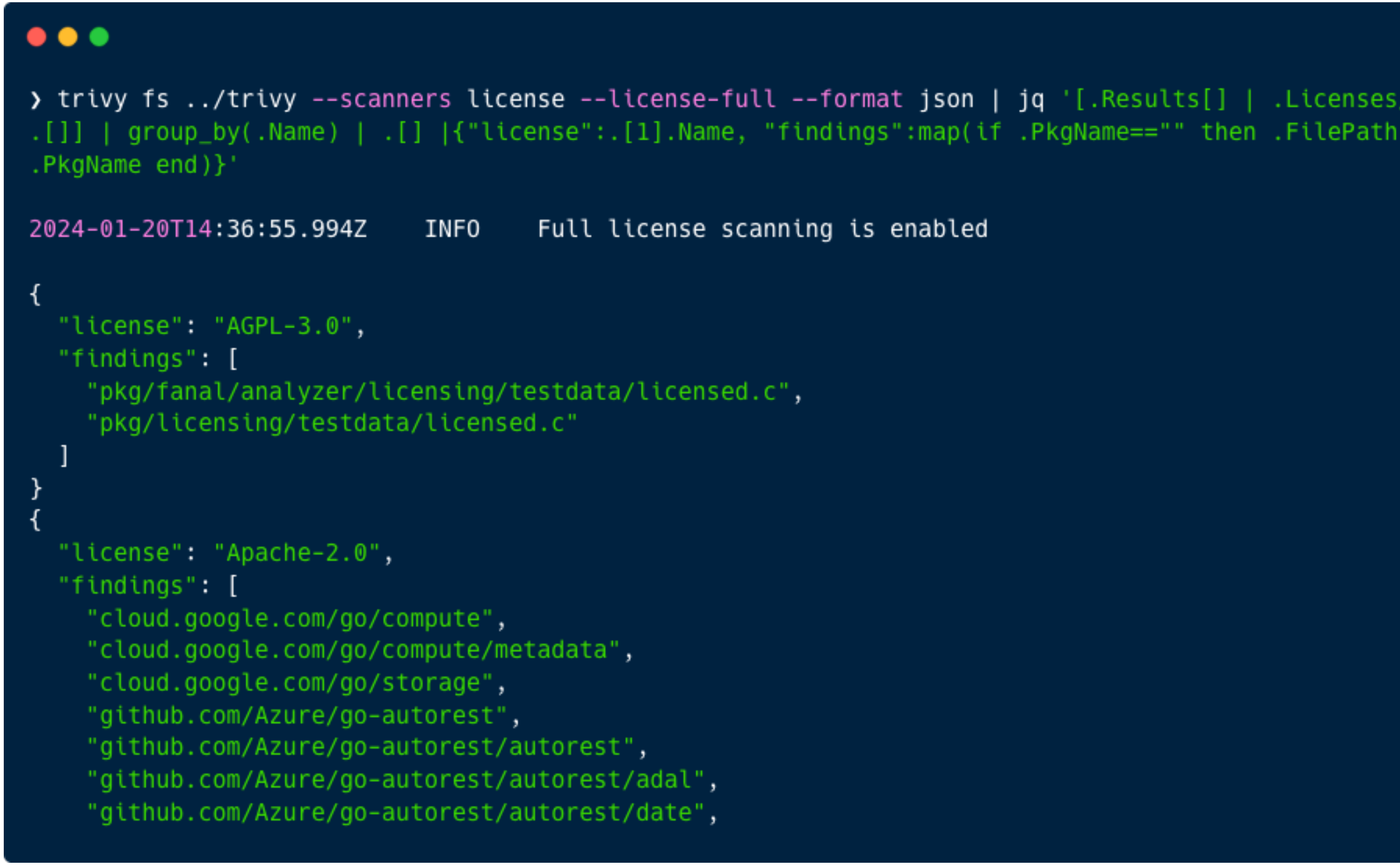
```
trivy image --severity HIGH,CRITICAL --vuln-type os postgres:10.6

/etc/ssl/private/ssl-cert-snakeoil.key (secrets)

Total: 1 (HIGH: 1, CRITICAL: 0)

HIGH: AsymmetricPrivateKey (private-key)
═══════════════════════════════════════════════════════════════════════════════════════════════════════════════════════
═════════════════════════════════════════════════════════════════════════════════════════
Asymmetric Private Key
───────────────────────────────────────────────────────────────────────────────────────────────────────────────────────
─────────────────────────────────────────────────────────────────────────────────────────
 /etc/ssl/private/ssl-cert-snakeoil.key:1 (added by 'set -ex;       export PYTHONDONTWRITEBYTECOD')
───────────────────────────────────────────────────────────────────────────────────────────────────────────────────────
─────────────────────────────────────────────────────────────────────────────────────────
   1 [ -----BEGIN PRIVATE
KEY-----***************************************************************************************************************
***********************************************************************************************************************
***********************************************************************************************************************
***********************************************************************************************************************
***********************************************************************************************************************
***********************************************************************************************************************
***********************************************************************************************************************
***********************************************************************************************************************
***********************************************************************************************************************
***********************************************************************************************************************
***********************************************************************************************************************
***********************************************************************************************************************
***********************************************************************************************************************
************-----END PRIVATE KEY-----
   2
───────────────────────────────────────────────────────────────────────────────────────────────────────────────────────
────────────────────────────────────────────────────────────────────────────────────────
```

License Scanning

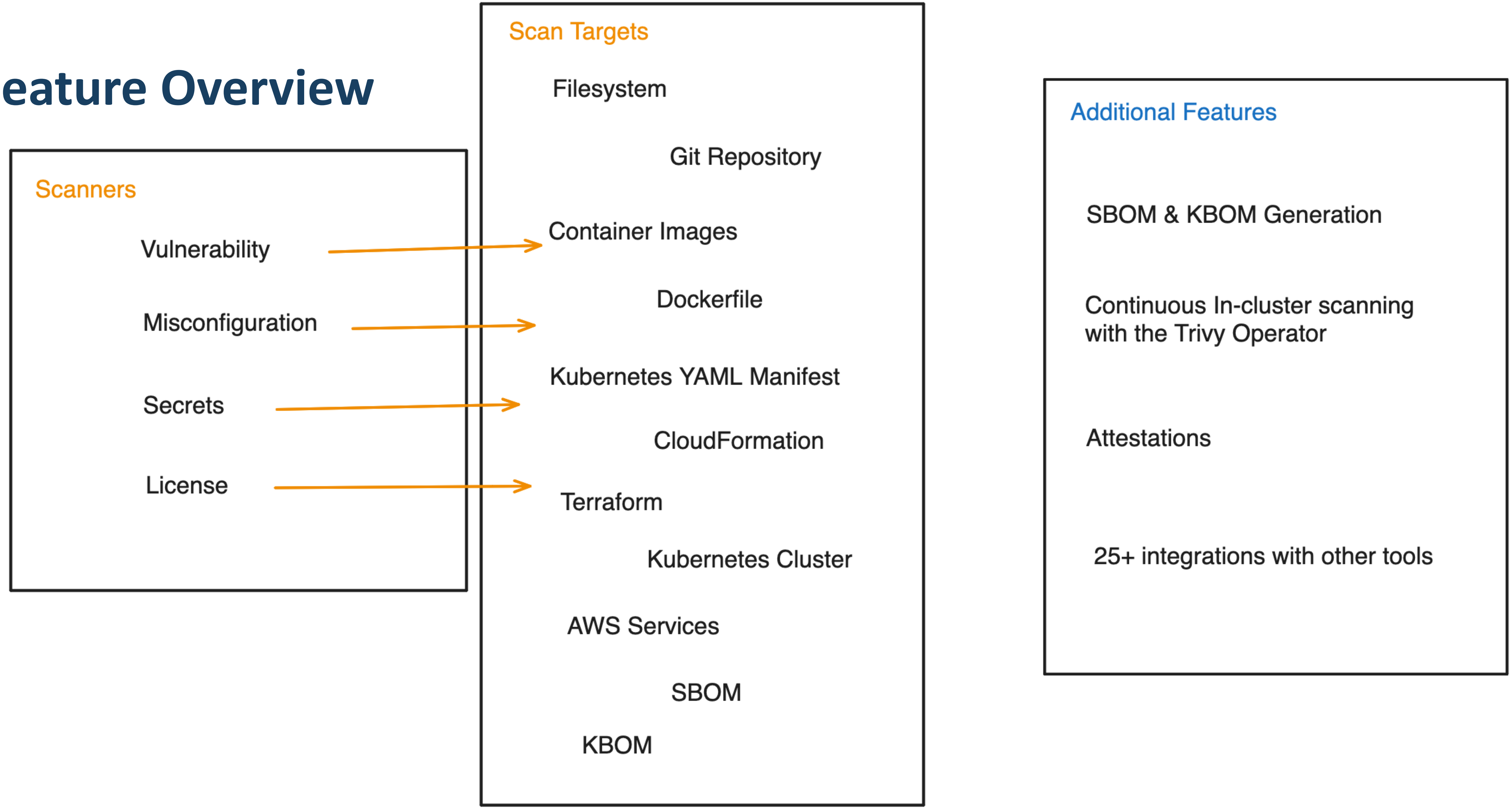


```
❯ trivy fs ../trivy --scanners license --license-full --format json | jq '[.Results[] | .Licenses
.[]] | group_by(.Name) | .[] |{"license":.[1].Name, "findings":map(if .PkgName=="" then .FilePath
.PkgName end)}'

2024-01-20T14:36:55.994Z    INFO    Full license scanning is enabled

{
  "license": "AGPL-3.0",
  "findings": [
    "pkg/fanal/analyzer/licensing/testdata/licensed.c",
    "pkg/licensing/testdata/licensed.c"
  ]
}
{
  "license": "Apache-2.0",
  "findings": [
    "cloud.google.com/go/compute",
    "cloud.google.com/go/compute/metadata",
    "cloud.google.com/go/storage",
    "github.com/Azure/go-autorest",
    "github.com/Azure/go-autorest/autorest",
    "github.com/Azure/go-autorest/autorest/adal",
    "github.com/Azure/go-autorest/autorest/date",
```

Feature Overview



Scanners

- Vulnerability
- Misconfiguration
- Secrets
- License

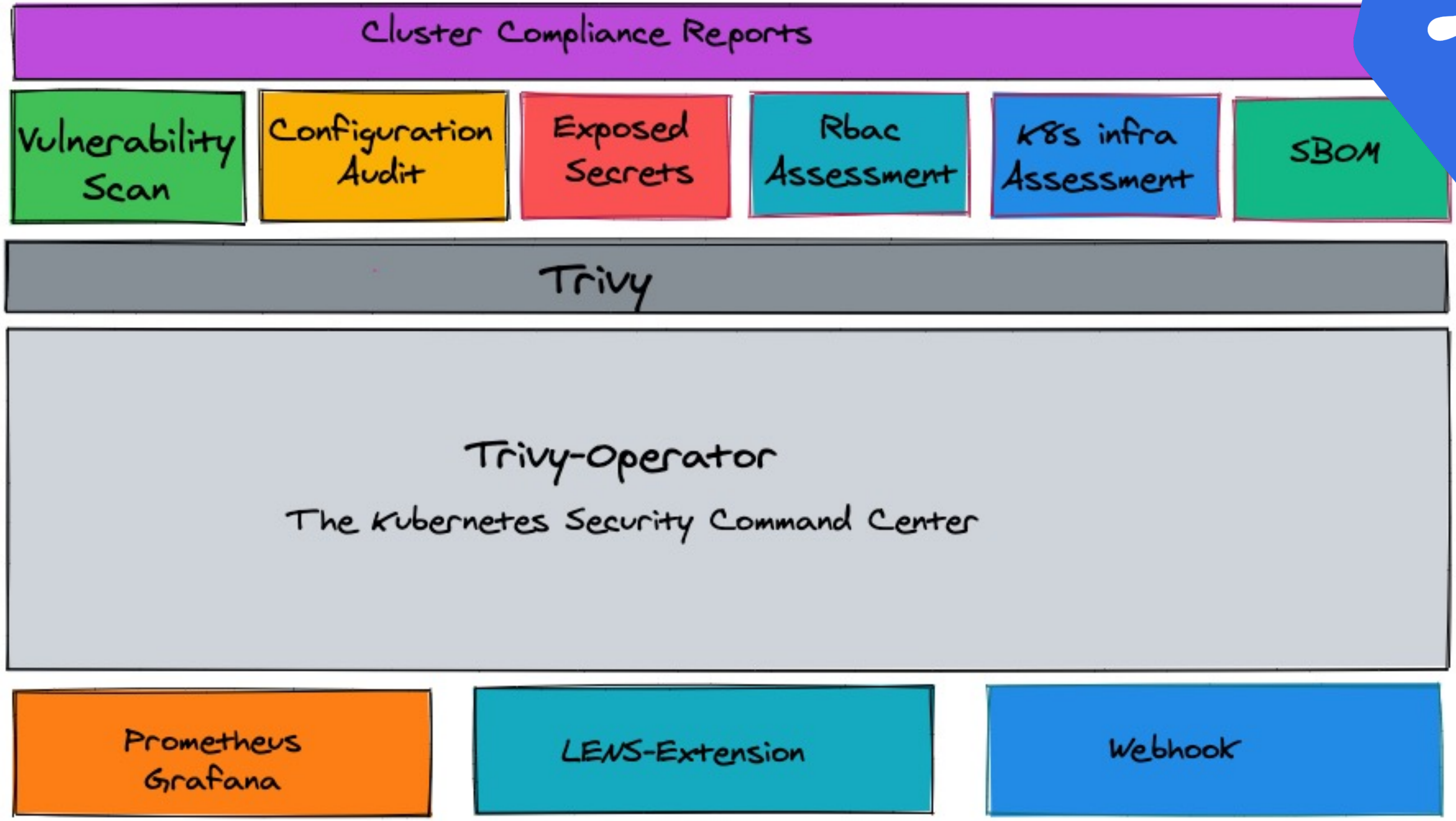
Scan Targets

- Filesystem
- Git Repository
- Container Images
- Dockerfile
- Kubernetes YAML Manifest
- CloudFormation
- Terraform
- Kubernetes Cluster
- AWS Services
- SBOM
- KBOM

Additional Features

- SBOM & KBOM Generation
- Continuous In-cluster scanning with the Trivy Operator
- Attestations
- 25+ integrations with other tools

The Trivy Operator



GitHub Repository

Specific Demos/Guides

- Trivy AWS Service Scanning
- Trivy Kyverno and Cosign
- Trivy KBOM and Kubernetes resources scanning

Additional Resources

Trivy GitHub repository

Trivy Operator

Trivy demo repository

⭐ Give us a Star on GitHub ⭐