```
/Users/hrishikesh/Desktop/github_projects/vuln-reachability-sample/.venv/bin/python /Users/hrishikesh/Desktop/github_projects/vuln-reachability-sample/tracer_.py python_vuln_app --output-sbom project.sbom.json --output-report security_report.json --output-consolidated consolidated.json --run-reachability
   Starting Security Analysis with Syft and Trivy...
   Security findings will be saved to: security_findings/python_vuln_app
   Syft version: Application:   syft
Version:       1.30.0
BuildDate:     2025-08-08T18:21:45Z
GitCommit:     49736e7c4acd3cb1aed2d3a8c8cdcd696ac77166
GitDescription: v1.30.0
Platform:      darwin/arm64
GoVersion:     go1.24.5
Compiler:      gc
SchemaVersion: 16.0.36
  ✓ Trivy version: Version: 0.65.0
Vulnerability DB:
  Version: 2
  UpdatedAt: 2025-08-15 12:30:05.636607223 +0000 UTC
  NextUpdate: 2025-08-16 12:30:05.636606962 +0000 UTC
  DownloadedAt: 2025-08-15 15:35:50.895025 +0000 UTC

   Generating SBOM from: python_vuln_app
   Generating SBOM with Syft for: python_vuln_app
   SBOM generated successfully: security_findings/python_vuln_app/project.sbom.json
   Parsed 21 components from SBOM
```

```
   🔍 Scanning SBOM with Trivy: security_findings/python_vuln_app/project.sbom.json
    Found 45 vulnerabilities

    Generating security report...

================================================================================
   🔒  SECURITY SCAN RESULTS
================================================================================
    Scan completed at: 2025-08-15T21:34:56.126100
    SBOM Generator: Syft
    Vulnerability Scanner: Trivy

    Total Components: 21
⚠️🔍  Vulnerable Components: 11
    Total Vulnerabilities: 45

    Severity Breakdown:
       CRITICAL: 3
       HIGH: 14
       MEDIUM: 23
       LOW: 5

    TOP CRITICAL/HIGH VULNERABILITIES:
----------------------------------------------------------
    CVE-2023-37920 - certifi@2020.12.5
    Severity: HIGH (CVSS: 7.5)
    Title: python-certifi: Removal of e-Tugra root certificate
```

```
51       Fixed in: 2023.7.22
52
53    CVE-2023-0286 - cryptography@3.4.8
54    Severity: HIGH (CVSS: 7.4)
55    Title: openssl: X.400 address type confusion in X.509 GeneralName
56       Fixed in: 39.0.1
57
58    CVE-2023-50782 - cryptography@3.4.8
59    Severity: HIGH (CVSS: 7.5)
60    Title: python-cryptography: Bleichenbacher timing oracle attack against RSA
   decryption - incomplete fix for CVE-2020-25659
61       Fixed in: 42.0.0
62
63    CVE-2023-30861 - flask@2.0.1
64    Severity: HIGH (CVSS: 7.5)
65    Title: flask: Possible disclosure of permanent session cookie due to missing
   Vary: Cookie header
66       Fixed in: 2.3.2, 2.2.5
67
68    CVE-2021-34552 - pillow@8.2.0
69    Severity: CRITICAL (CVSS: 9.8)
70    Title: python-pillow: Buffer overflow in image convert function
71       Fixed in: 8.3.0
72
73    CVE-2022-22817 - pillow@8.2.0
74    Severity: CRITICAL (CVSS: 9.8)
75    Title: python-pillow: PIL.ImageMath.eval allows evaluation of arbitrary
```

```
expressions
      Fixed in: 9.0.1

   CVE-2023-50447 - pillow@8.2.0
   Severity: CRITICAL (CVSS: 8.1)
   Title: pillow: Arbitrary Code Execution via the environment parameter
      Fixed in: 10.2.0

   CVE-2021-23437 - pillow@8.2.0
   Severity: HIGH (CVSS: 7.5)
   Title: python-pillow: possible ReDoS via the getrgb function
      Fixed in: 8.3.2

   CVE-2022-24303 - pillow@8.2.0
   Severity: HIGH (CVSS: 9.1)
   Title: python-pillow: temporary directory with a space character allows
  removal of unrelated file after im.show() and related actions
      Fixed in: 9.0.1

   CVE-2022-45198 - pillow@8.2.0
   Severity: HIGH (CVSS: 7.5)
   Title: Pillow before 9.2.0 performs Improper Handling of Highly Compressed GI
    ...
      Fixed in: 9.2.0

⚠  Please review and remediate the vulnerabilities above.
================================================================================
```

```
100
101    Full report saved to: security_findings/python_vuln_app/security_report.json
102    Consolidated recommendations saved to: security_findings/python_vuln_app/
    consolidated.json
103    Consolidated recommendations saved to: security_findings/python_vuln_app/
    consolidated.json
104
105    Running vulnerability reachability analysis...
106 === Vulnerability Reachability Analysis ===
107 Total vulnerabilities analyzed: 11
108 Critical (actively used): 0
109 High (used with calls): 2
110 Medium (imported): 0
111 Low (limited usage): 0
112 Not reachable: 9
113
114    HIGH: flask v2.0.1
115    Reason: Package flask is actively used with direct function calls
116    Upgrade to: 2.3.2, 2.2.5
117       python_vuln_app/src/app.py:10 - from flask import Flask, request,
    render_template_string, jsonify
118       python_vuln_app/src/app.py:16 - app = Flask(__name__)
119
120    HIGH: requests v2.25.1
121    Reason: Package requests is actively used with direct function calls
122    Upgrade to: 2.32.4
123       python_vuln_app/src/app.py:11 - import requests
```

```
124       python_vuln_app/src/app.py:81 - response = requests.get('https://httpbin.
    org/get', timeout=5)
125
126    Reachability analysis saved to: security_findings/python_vuln_app/
    vulnerability_reachability_report.json
127
128    Found 17 CRITICAL/HIGH vulnerabilities!
129
130 Process finished with exit code 1
131
```