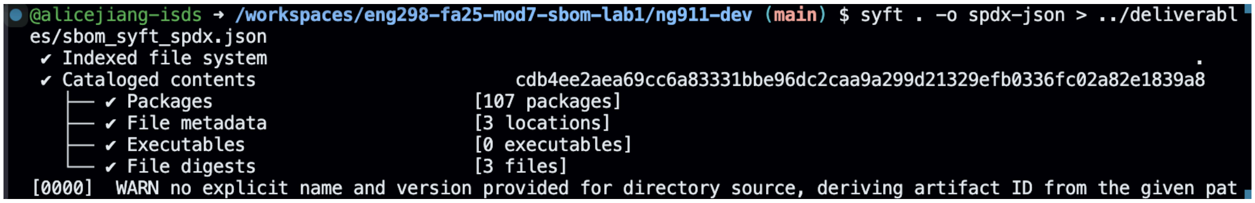
Alice Jiang
Foundational Technical & Organizational Concepts & Practices in Cybersecurity 298
Professor O'Brien
12/08/25

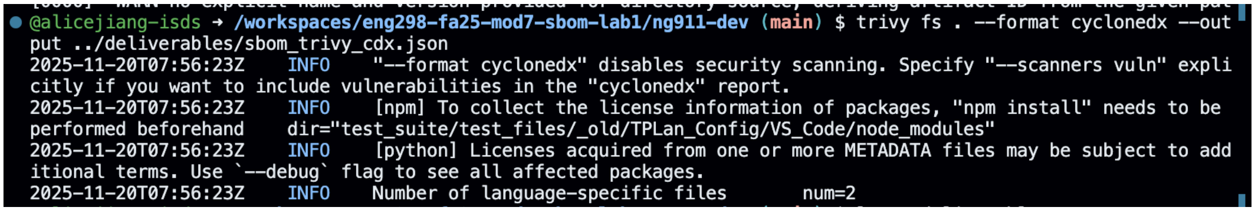
Module 7: Assignment: Secure Software Supply Chain with SBOMs

1. How many components each tool reported (Syft vs. Trivy)
    a. For Syft, it has 107 packages, making it 107 components



```
@alicejiang-isds ➜ /workspaces/eng298-fa25-mod7-sbom-lab1/ng911-dev (main) $ syft . -o spdx-json > ../deliverabl
es/sbom_syft_spdx.json
 ✔ Indexed file system                                                                                         .
 ✔ Cataloged contents                              cdb4ee2aea69cc6a83331bbe96dc2caa9a299d21329efb0336fc02a82e1839a8
   ├── ✔ Packages                             [107 packages]
   ├── ✔ File metadata                        [3 locations]
   ├── ✔ Executables                          [0 executables]
   └── ✔ File digests                         [3 files]
[0000]  WARN no explicit name and version provided for directory source, deriving artifact ID from the given pat
```

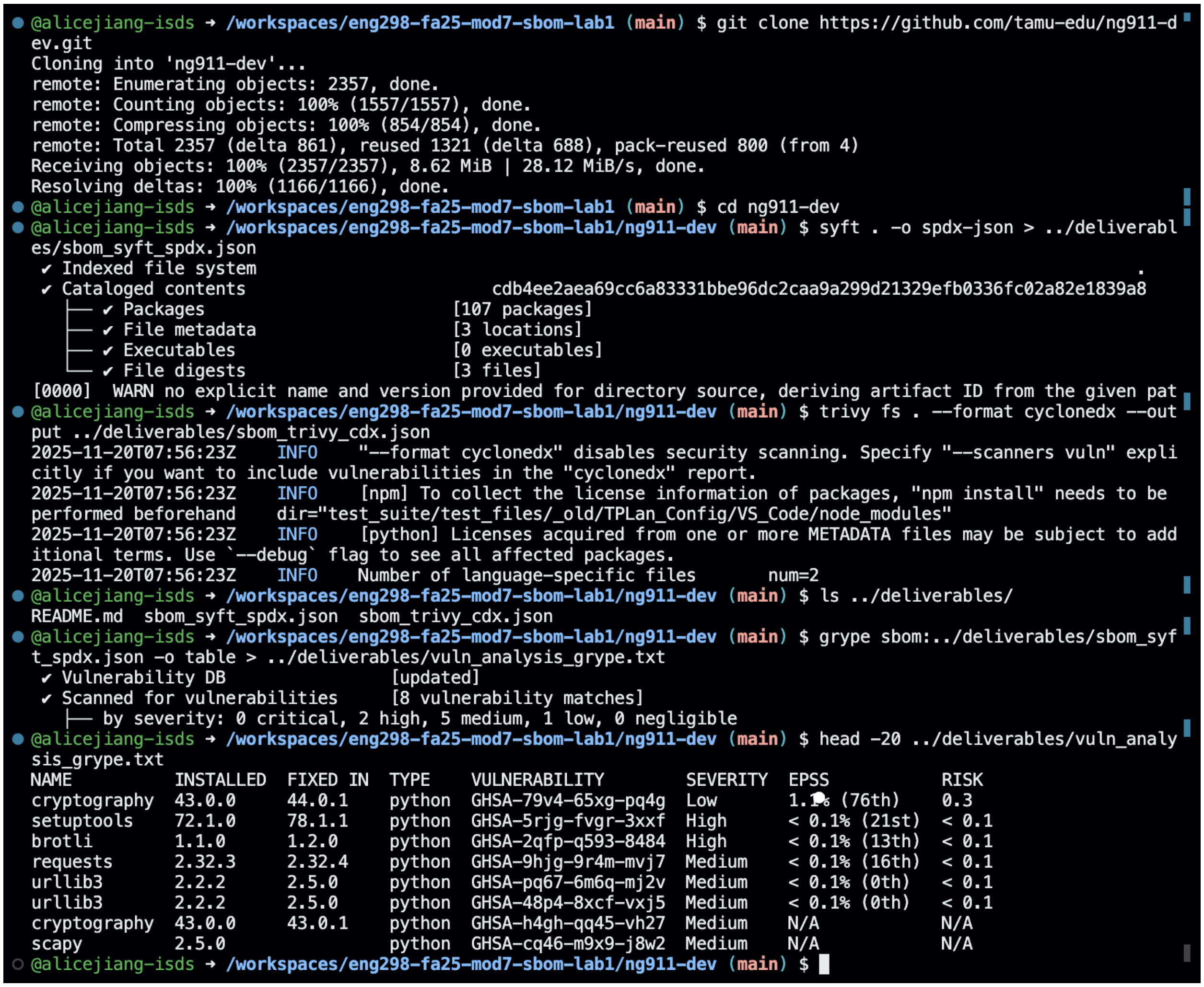
    b. For Trivy, there's 2 language-specific files, making it 2 components



```
@alicejiang-isds ➜ /workspaces/eng298-fa25-mod7-sbom-lab1/ng911-dev (main) $ trivy fs . --format cyclonedx --out
put ../deliverables/sbom_trivy_cdx.json
2025-11-20T07:56:23Z    INFO    "--format cyclonedx" disables security scanning. Specify "--scanners vuln" expli
citly if you want to include vulnerabilities in the "cyclonedx" report.
2025-11-20T07:56:23Z    INFO    [npm] To collect the license information of packages, "npm install" needs to be
performed beforehand    dir="test_suite/test_files/_old/TPLan_Config/VS_Code/node_modules"
2025-11-20T07:56:23Z    INFO    [python] Licenses acquired from one or more METADATA files may be subject to add
itional terms. Use `--debug` flag to see all affected packages.
2025-11-20T07:56:23Z    INFO    Number of language-specific files       num=2
```

2. One difference you notice between the SPDX SBOM and the CycloneDX SBOM (format, fields, component count, etc.)
    a. One difference I notice between SPDX SBOM and the CycloneDX SBOM is that the SPDX SBOM is more organized in the format, it tells you the number of packages, file digests, file metadata, and executables there are, and the count is shown on the right side. While for CycloneDX SBOM, it's more compact in the format, there's no table-like field, instead it's separated into lines, but it shows the count, just within a line.

Here is the screenshot of the whole code for the lab:



```
● @alicejiang-isds → /workspaces/eng298-fa25-mod7-sbom-lab1 (main) $ git clone https://github.com/tamu-edu/ng911-d
ev.git
Cloning into 'ng911-dev'...
remote: Enumerating objects: 2357, done.
remote: Counting objects: 100% (1557/1557), done.
remote: Compressing objects: 100% (854/854), done.
remote: Total 2357 (delta 861), reused 1321 (delta 688), pack-reused 800 (from 4)
Receiving objects: 100% (2357/2357), 8.62 MiB | 28.12 MiB/s, done.
Resolving deltas: 100% (1166/1166), done.
● @alicejiang-isds → /workspaces/eng298-fa25-mod7-sbom-lab1 (main) $ cd ng911-dev
● @alicejiang-isds → /workspaces/eng298-fa25-mod7-sbom-lab1/ng911-dev (main) $ syft . -o spdx-json > ../deliverabl
es/sbom_syft_spdx.json
 ✔ Indexed file system                                                                        .
 ✔ Cataloged contents                         cdb4ee2aea69cc6a83331bbe96dc2caa9a299d21329efb0336fc02a82e1839a8
   ├── ✔ Packages                        [107 packages]
   ├── ✔ File metadata                   [3 locations]
   ├── ✔ Executables                     [0 executables]
   └── ✔ File digests                    [3 files]
[0000]  WARN no explicit name and version provided for directory source, deriving artifact ID from the given pat
● @alicejiang-isds → /workspaces/eng298-fa25-mod7-sbom-lab1/ng911-dev (main) $ trivy fs . --format cyclonedx --out
put ../deliverables/sbom_trivy_cdx.json
2025-11-20T07:56:23Z    INFO    "--format cyclonedx" disables security scanning. Specify "--scanners vuln" expli
citly if you want to include vulnerabilities in the "cyclonedx" report.
2025-11-20T07:56:23Z    INFO    [npm] To collect the license information of packages, "npm install" needs to be
performed beforehand    dir="test_suite/test_files/_old/TPLan_Config/VS_Code/node_modules"
2025-11-20T07:56:23Z    INFO    [python] Licenses acquired from one or more METADATA files may be subject to add
itional terms. Use `--debug` flag to see all affected packages.
2025-11-20T07:56:23Z    INFO    Number of language-specific files       num=2
● @alicejiang-isds → /workspaces/eng298-fa25-mod7-sbom-lab1/ng911-dev (main) $ ls ../deliverables/
README.md  sbom_syft_spdx.json  sbom_trivy_cdx.json
● @alicejiang-isds → /workspaces/eng298-fa25-mod7-sbom-lab1/ng911-dev (main) $ grype sbom:../deliverables/sbom_syf
t_spdx.json -o table > ../deliverables/vuln_analysis_grype.txt
 ✔ Vulnerability DB                [updated]
 ✔ Scanned for vulnerabilities     [8 vulnerability matches]
   ├── by severity: 0 critical, 2 high, 5 medium, 1 low, 0 negligible
● @alicejiang-isds → /workspaces/eng298-fa25-mod7-sbom-lab1/ng911-dev (main) $ head -20 ../deliverables/vuln_analy
sis_grype.txt
NAME          INSTALLED  FIXED IN  TYPE    VULNERABILITY        SEVERITY  EPSS           RISK
cryptography  43.0.0     44.0.1    python  GHSA-79v4-65xg-pq4g  Low       1.1% (76th)    0.3
setuptools    72.1.0     78.1.1    python  GHSA-5rjg-fvgr-3xxf  High      < 0.1% (21st)  < 0.1
brotli        1.1.0      1.2.0     python  GHSA-2qfp-q593-8484  High      < 0.1% (13th)  < 0.1
requests      2.32.3     2.32.4    python  GHSA-9hjg-9r4m-mvj7  Medium    < 0.1% (16th)  < 0.1
urllib3       2.2.2      2.5.0     python  GHSA-pq67-6m6q-mj2v  Medium    < 0.1% (0th)   < 0.1
urllib3       2.2.2      2.5.0     python  GHSA-48p4-8xcf-vxj5  Medium    < 0.1% (0th)   < 0.1
cryptography  43.0.0     43.0.1    python  GHSA-h4gh-qq45-vh27  Medium    N/A            N/A
scapy         2.5.0                python  GHSA-cq46-m9x9-j8w2  Medium    N/A            N/A
○ @alicejiang-isds → /workspaces/eng298-fa25-mod7-sbom-lab1/ng911-dev (main) $ 
```

3. Copy the top 5 rows into your report table. Then select one CVE, locate it in the NVD Database, and summarize its cause or impact in one sentence.

NAME   INSTALLED FIXED IN TYPE   VULNERABILITY   SEVERITY EPSS   RISK
cryptography 43.0.0   44.0.1   python GHSA-79v4-65xg-pq4g Low   1.1% (76th)   0.3
setuptools   72.1.0   78.1.1   python GHSA-5rjg-fvgr-3xxf High   < 0.1% (21st) < 0.1
brotli   1.1.0   1.2.0   python GHSA-2qfp-q593-8484 High   < 0.1% (13th) < 0.1
requests   2.32.3   2.32.4   python GHSA-9hjg-9r4m-mvj7 Medium   < 0.1% (16th) < 0.1
urllib3   2.2.2   2.5.0   python GHSA-pq67-6m6q-mj2v Medium   < 0.1% (0th)   < 0.1

NAME	INSTALLED	FIXED IN	TYPE	VULERNABILITY	SEVERITY	EPSS	RISK
cryptography	43.0.0	44.0.1	python	GHSA-79v4-65xg-pq4g	Low	1.1% (76th)	0.3

setuptools	72.1.0	78.1.1	python	GHSA-5rjg-fvgr-3xxf	High	< 0.1% (22st)	< 0.1
brotli	1.1.0	1.2.0	python	GHSA-2qfp-q593-8484	High	< 0.1% (13th)	< 0.1
requests	2.32.3	2.32.4	python	GHSA-9hjg-9r4m-mvj7	Medium	< 0.1% (16th)	< 0.1
urlib3	2.2.2	2.5.0	python	GHSA-pq67-6m6q-mj2v	Medium	< 0.1% (0th)	< 0.1

I had to use this website to translate the GHSA IDs to CVE: https://github.com/advisories ↴

NAME	INSTALLED	FIXED IN	TYPE	CVE	SEVERITY	EPSS	RISK
cryptography	43.0.0	44.0.1	python	CVE-2024-12797	Low	1.1% (76th)	0.3
setuptools	72.1.0	78.1.1	python	CVE-2025-47273	High	< 0.1% (22st)	< 0.1
brotli	1.1.0	1.2.0	python	CVE-2025-6176	High	< 0.1% (13th)	< 0.1
requests	2.32.3	2.32.4	python	CVE-2024-47081	Medium	< 0.1% (16th)	< 0.1
urlib3	2.2.2	2.5.0	python	CVE-2025-50181	Medium	< 0.1% (0th)	< 0.1

The CVE I chose is this: CVE-2025-47273

The impact of this specific CVE is that there was a path traversal vulnerability in PackageIndex of the package setuptools, where attackers can write files to arbitrary locations on the filesystem with permission on Python code, which means they can do remote code execution——luckily this got fixed in version 78.1.1.