INL/CON-21-64718-Revision-0

Energysec: Generating SBoMs Utilizing Structured Threat Information Expression JSON bundles

October 2021

Changing the World's Energy Future

Manuel Vazquez Jr.

Idaho National Laboratory

INL/CON-21-64718-Revision-0

Energysec: Generating SBoMs Utilizing Structured Threat Information Expression JSON bundles

Manuel Vazquez Jr.

October 2021

Idaho National Laboratory
Idaho Falls, Idaho 83415

http://www.inl.gov

Prepared for the
U.S. Department of Energy
Under DOE Idaho Operations Office
Contract DE-AC07-05ID14517

Generating SBoMs Utilizing Structured Threat Information Expression JSON bundles

Idaho National Laboratory – Manuel Vazquez

Agenda

- Introduction
- STIX Overview
- Project Overview
- Software Bill of Materials
- Questions

STIX™

Who am I?

Introduction

- Software Developer/Researcher with Idaho National Laboratory for about 1 year.
- BS Information Systems Technology Cybersecurity / Computer Science: California State University, San Bernardino
- I enjoy ‘breaking’ things.

STIX

- What is STIX?
  - A format used to share and serialize using XML (STIX v1)/ JSON (STIX v2)
  - STIX uses its own syntax
  - Has committee that manages specification
  - Designed to improve capabilities in:
    - Automated threat exchange
    - Automated detection and response
    - Collaborative threat analysis

  https://oasis-open.github.io/cti-documentation

FC2: Firmware Command and Control

Firmware Command and Control (FC2)

- *Funding from* Department of Energy Offices
- Energy Efficiency and Renewable Energy (EERE) – Solar Energy Technology Office (SETO) Guohui Yuan
- Cybersecurity, Energy Security and Emergency Response (CESER) Akhlesh Kaushiva (AK)
- EERE - Building Technology Office (BTO) Erika Gupta

INL Lead with Laboratory Partners:

National Renewable Energy Laboratory (Maurice Martin)

Sandia National Laboratory (Chris Lamb)

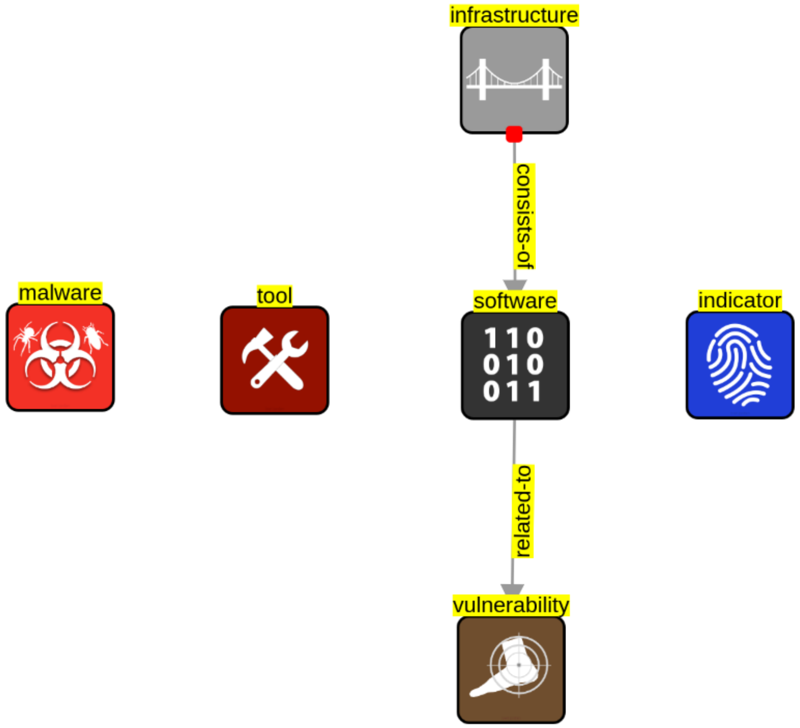
Argonne National Laboratory (Hyekyung (Clarisse) Kim)

Industry Partners:

Project FC2: Embedded Systems and Software Component Results Translated to STIX

Mapping components into STIX

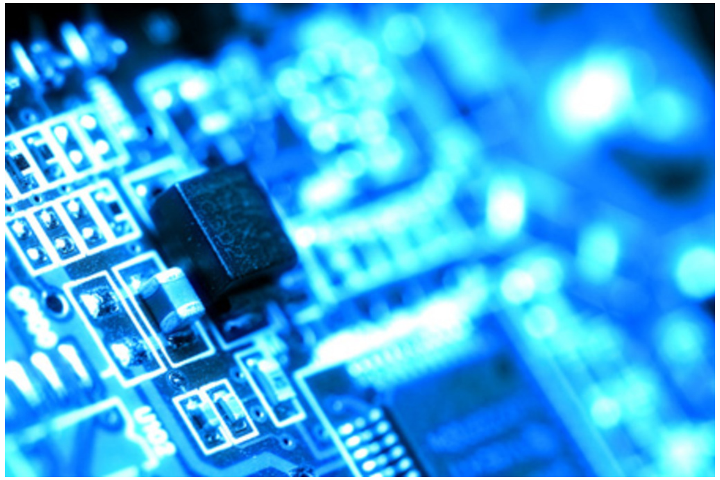
- Some of the components that we were able to map:
  - Hardware
  - The Flow of Data
  - Process, File Trees, Network Topology, Attack Surfaces via NVD scripts or OpenVAS and **Software Bill of Materials.**
- We leverage scripts to translate software component and get results

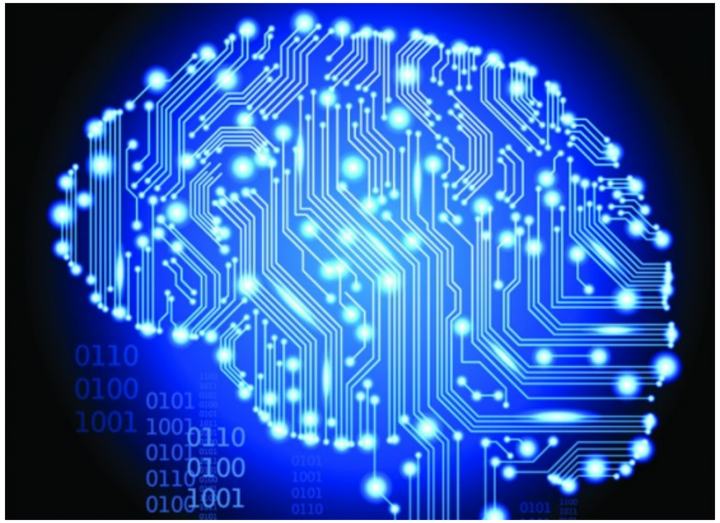


Software Bill of Materials

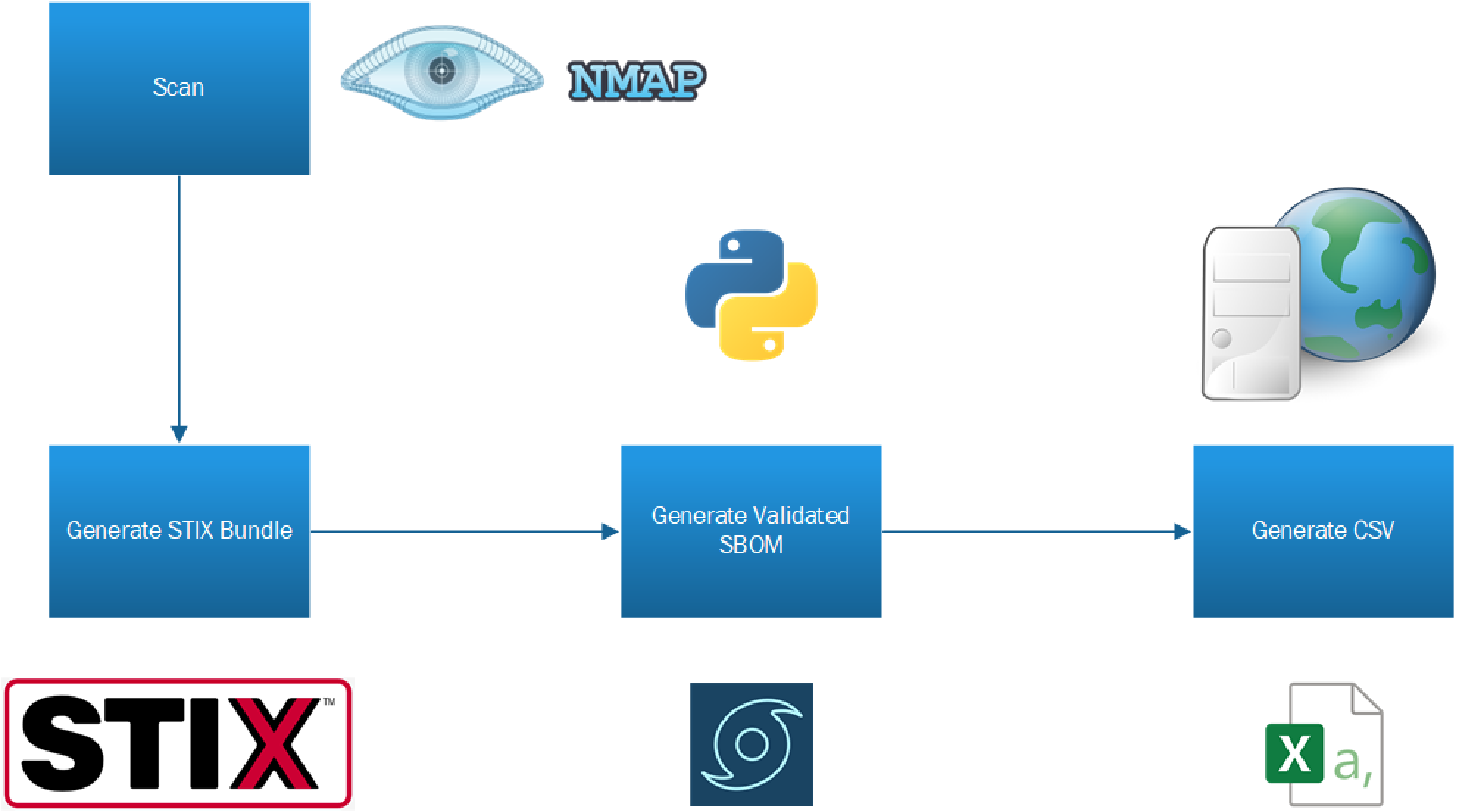
What is this phenomenon?

- List of components in a piece of software.
- Software Bill of Materials aims to fix this by providing information to the end user.
- There are Different formats:
  - CycloneDX
  - Software Package Data Exchange (SPDX)
  - Software Identification (SWID) Tags
- This can be used to identify vulnerabilities or license issues.
- In the Cyber world, information is key. Here at Idaho National Laboratory, we are seeking ways to solve this issue to help identify potential issues ahead of time.

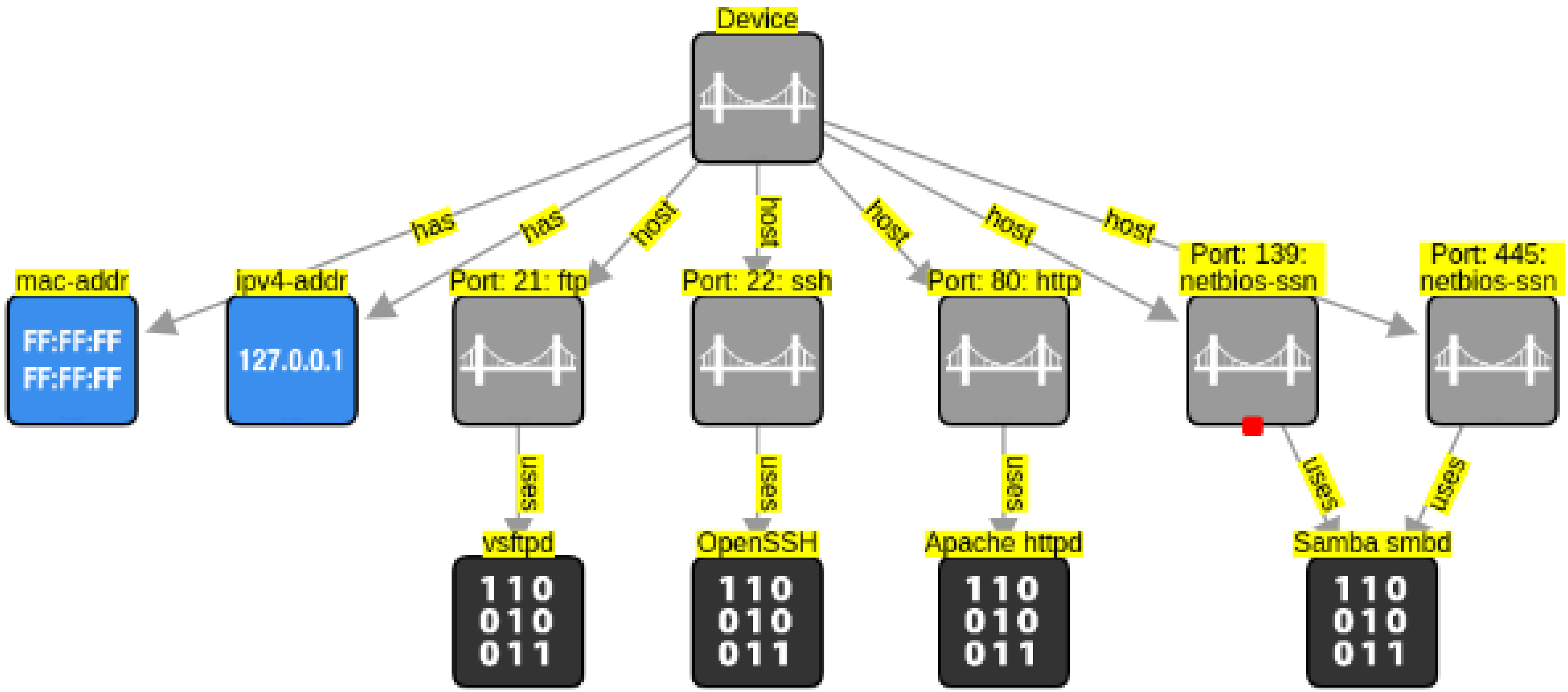




INL SBOM Generation



IX: Mapping our Applications



Generating SBOM from STIX Bundles

- Utilizing Python we're able to extract application from bundles to generate SBOM
- There are different formats, this example can search for software within your environment and generate a CycloneDX format Software Bill of Material.


```
{
    "bomFormat": "CycloneDX",
    "specVersion": "1.3",
    "serialNumber": "urn:uuid:3e671687-395b-41f5-a30f-a58921a69b79",
    "version": 1,
    "components": [
        {
            "type": "operating-system",
            "name": "Linux 2.6.9 - 2.6.33",
            "version": "Linux 2.6.9 - 2.6.33",
            "cpe": "cpe:/o:linux:linux_kernel:2.6"
        },
        {
            "type": "application",
            "name": "vsftpd",
            "version": "2.3.4",
            "cpe": "cpe:/a:vsftpd:vsftpd:2.3.4"
        },
        {
            "type": "application",
            "name": "OpenSSH",
            "version": "4.7p1 Debian 8ubuntu1",
            "cpe": "cpe:/o:linux:linux_kernel"
        },
        {
            "type": "application",
            "name": "Linux telnetd",
            "version": "",
            "cpe": "cpe:/o:linux:linux_kernel"
        },
        {
            "type": "application",
            "name": "Postfix smtpd",
            "version": "",
            "cpe": "cpe:/a:postfix:postfix"
        },
        {
            "type": "application",
            "name": "ISC BIND",
            "version": "9.4.2",
            "cpe": "cpe:/a:isc:bind:9.4.2"
        },
```

	type	name	version	cpe
0	operating-system	Linux 2.6.9 - 2.6.33	Linux 2.6.9 - 2.6.33	cpe:/o:linux:linux_kernel:2.6
1	application	vsftpd	2.3.4	cpe:/a:vsftpd:vsftpd:2.3.4
2	application	OpenSSH	4.7p1 Debian 8ubuntu1	cpe:/o:linux:linux_kernel
3	application	Linux telnetd		cpe:/o:linux:linux_kernel
4	application	Postfix smtpd		cpe:/a:postfix:postfix
5	application	ISC BIND	9.4.2	cpe:/a:isc:bind:9.4.2
6	application	Apache httpd	2.2.8	cpe:/a:apache:http_server:2.2.8
7	application		2	
8	application	Samba smbd	3.X - 4.X	cpe:/a:samba:samba
9	application	netkit-rsh rexecd		cpe:/o:linux:linux_kernel
10	application	OpenBSD or Solaris rlogind		
11	application	GNU Classpath grmiregistry		
12	application	Metasploitable root shell		
13	application	ProFTPD	1.3.1	cpe:/a:proftpd:proftpd:1.3.1
14	application	MySQL	5.0.51a-3ubuntu5	cpe:/a:mysql:mysql:5.0.51a-3ubuntu5
15	application	distccd	v1	
16	application	PostgreSQL DB	8.3.0 - 8.3.7	cpe:/a:postgresql:postgresql:8.3
17	application	VNC		
18	application	UnrealIRCd		cpe:/a:unrealircd:unrealircd
19	application	Ruby DRb RMI		cpe:/a:ruby-lang:ruby:1.8

Download

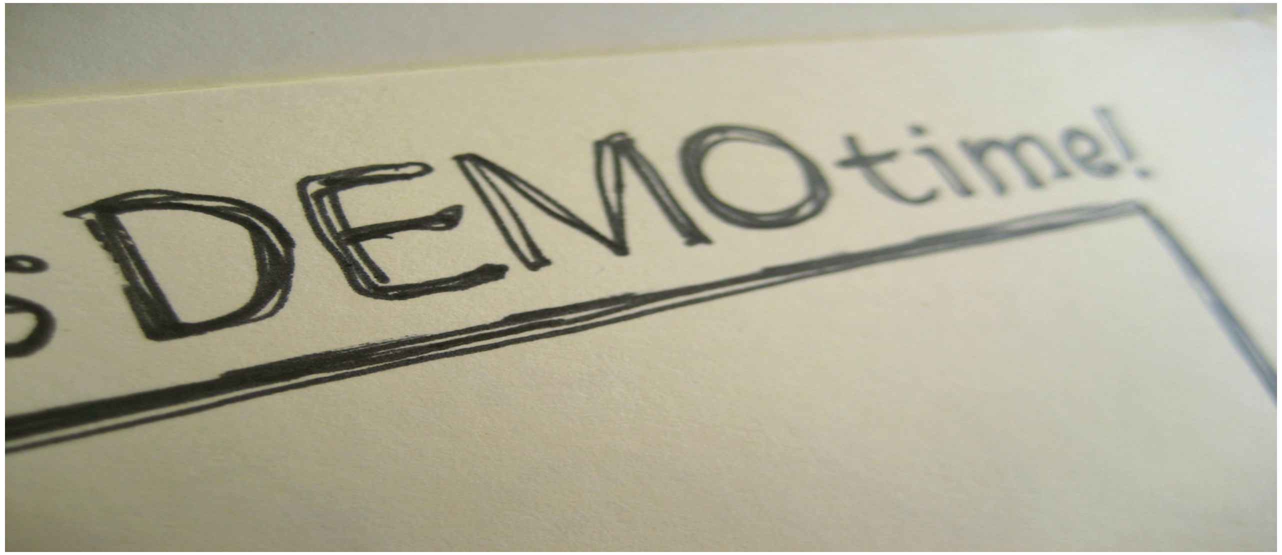
Potential Vulnerabilities

Vendor	Product	Version	Update	Edition	Language
cpe:/a:samba:samba:- View CVEs samba	samba	-			
cpe:/a:samba:samba:1.9.17 View CVEs samba	samba	1.9.17			
cpe:/a:samba:samba:1.9.17:p1 View CVEs samba	samba	1.9.17	p1		
cpe:/a:samba:samba:1.9.17:p2 View CVEs samba	samba	1.9.17	p2		
cpe:/a:samba:samba:1.9.17:p3 View CVEs samba	samba	1.9.17	p3		

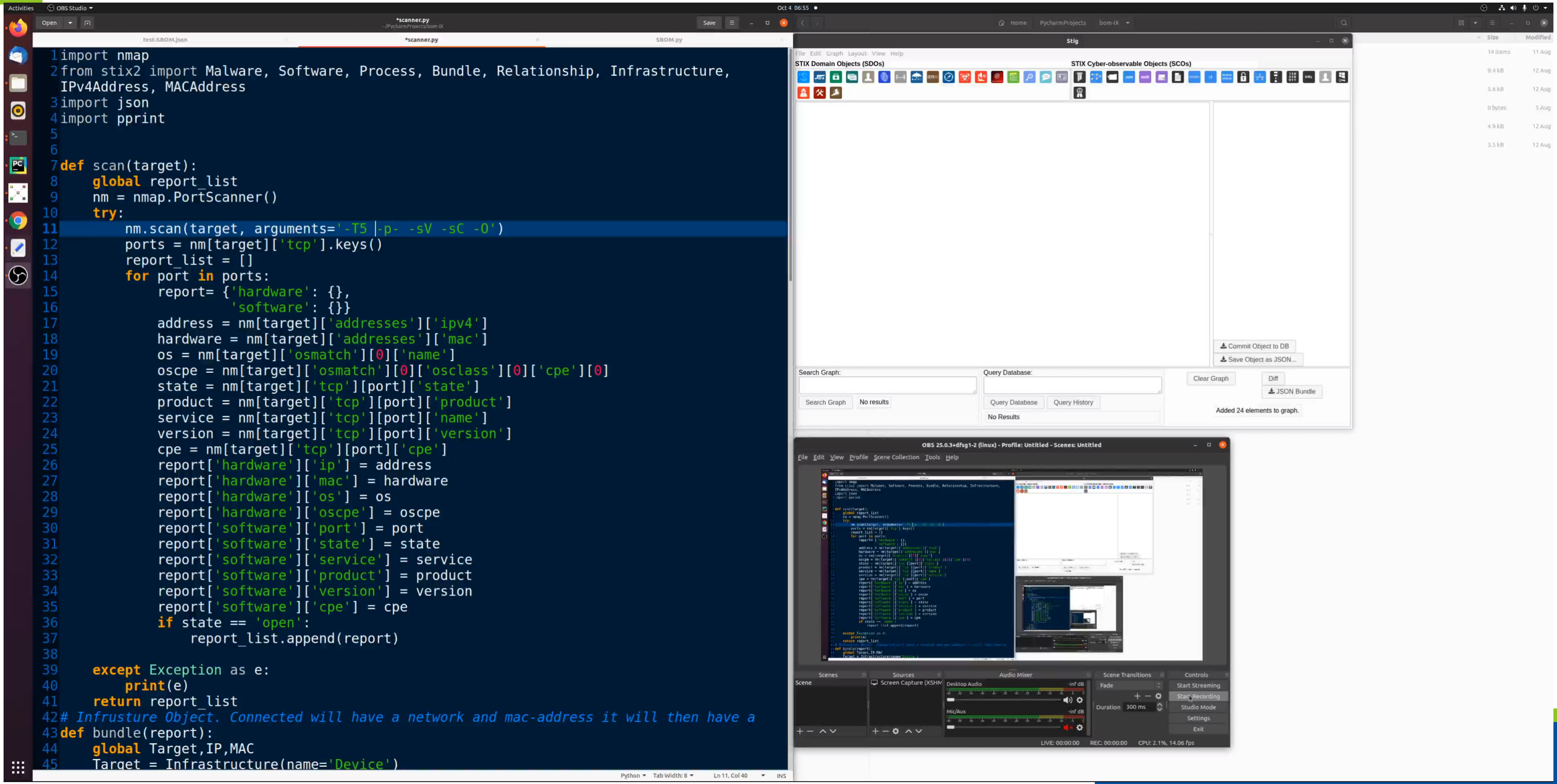
Vuln ID	Summary	CVSS Severity
CVE-2021-39246	Tor Browser through 10.5.6 and 11.x through 11.0a4 allows a correlation attack that can compromise the privacy of visits to v2 onion addresses. Exact timestamps of these onion-service visits are logged locally, and an attacker might be able to compare them to timestamp data collected by the destination server (or collected by a rogue site within the Tor network). **Published:** September 24, 2021; 3:15:07 PM -0400	V3.1: 6.1 MEDIUM V2.0: 3.6 LOW
CVE-2021-38877	IBM Jazz for Service Management 1.1.3.10 is vulnerable to stored cross-site scripting. This vulnerability allows users to embed arbitrary JavaScript code in the Web UI thus altering the intended functionality potentially leading to credentials disclosure within a trusted session. IBM X-Force ID: 208405. **Published:** September 23, 2021; 2:15:11 PM -0400	V3.1: 5.4 MEDIUM V2.0: 3.5 LOW
CVE-2021-29905	IBM Jazz for Service Management 1.1.3.10 and IBM Tivoli Netcool/OMNIbus_GUI is vulnerable to cross-site scripting. This vulnerability allows users to embed arbitrary JavaScript code in the Web UI thus altering the intended functionality potentially leading to credentials disclosure within a trusted session. IBM X-Force ID: 207616. **Published:** September 23, 2021; 2:15:11 PM -0400	V3.1: 5.4 MEDIUM V2.0: 3.5 LOW
CVE-2021-29904	IBM Jazz for Service Management 1.1.3.10 and IBM Tivoli Netcool/OMNIbus_GUI displays user credentials in plain clear text which can be read by a local user. IBM X-Force ID: 207610. **Published:** September 23, 2021; 2:15:11 PM -0400	V3.1: 5.5 MEDIUM V2.0: 2.1 LOW

Demo (Video)

- Video will show:
  - Generation of IX bundle
  - Generation of validated Software Bill of Material from IX Bundle
  - Use of Created web app to load the Software Bill of Materials
  - How this can be used to leverage potential vulnerabilities

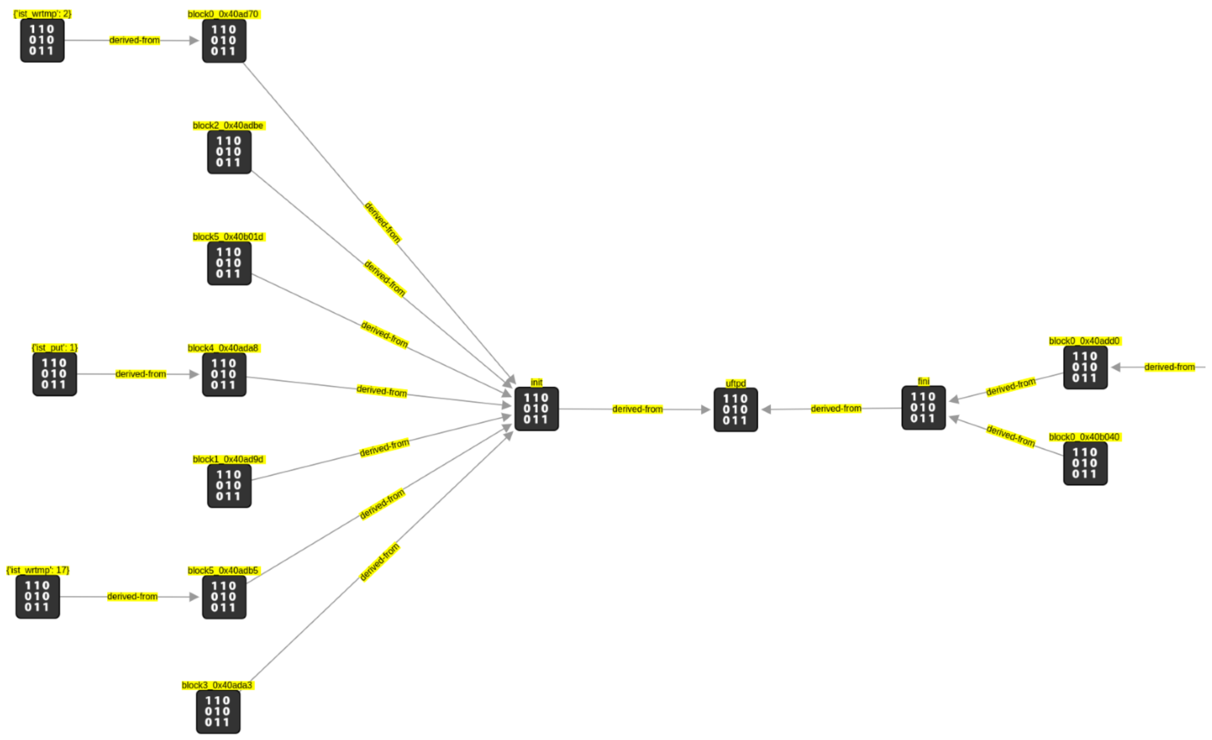


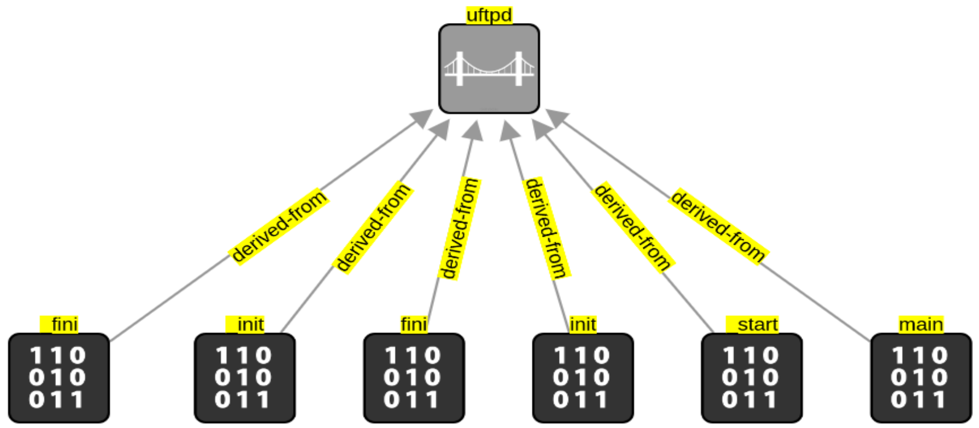
Video Demo


Activities
OBS Studio
Oct 4 06:55
Open
*scanner.py
Save
test-SBOM.json
*scanner.py
SBOM.py
import nmap
from stix2 import Malware, Software, Process, Bundle, Relationship, Infrastructure,
IPv4Address, MACAddress
import json
import pprint
def scan(target):
global report_list
nm = nmap.PortScanner()
try:
nm.scan(target, arguments='-T5 -p- -sV -sC -O')
ports = nm[target]['tcp'].keys()
report_list = []
for port in ports:
report= {'hardware': {},
'software': {}}
address = nm[target]['addresses']['ipv4']
hardware = nm[target]['addresses']['mac']
os = nm[target]['osmatch'][0]['name']
oscpe = nm[target]['osmatch'][0]['osclass'][0]['cpe'][0]
state = nm[target]['tcp'][port]['state']
product = nm[target]['tcp'][port]['product']
service = nm[target]['tcp'][port]['name']
version = nm[target]['tcp'][port]['version']
cpe = nm[target]['tcp'][port]['cpe']
report['hardware']['ip'] = address
report['hardware']['mac'] = hardware
report['hardware']['os'] = os
report['hardware']['oscpe'] = oscpe
report['software']['port'] = port
report['software']['state'] = state
report['software']['service'] = service
report['software']['product'] = product
report['software']['version'] = version
report['software']['cpe'] = cpe
if state == 'open':
report_list.append(report)
except Exception as e:
print(e)
return report_list
# Infrusture Object. Connected will have a network and mac-address it will then have a
def bundle(report):
global Target,IP,MAC
Target = Infrastructure(name='Device')
Python
Tab Width: 8
Ln 11, Col 40
INS
Home
PycharmProjects
bom-iX
Stig
File Edit Graph Layout View Help
STIX Domain Objects (SDOs)
STIX Cyber-observable Objects (SCOs)
Commit Object to DB
Save Object as JSON...
Search Graph:
Query Database:
Clear Graph
Diff
JSON Bundle
Search Graph
No results
Query Database
Query History
No Results
Added 24 elements to graph.
OBS 25.0.3+dfsg1-2 (linux) - Profile: Untitled - Scenes: Untitled
File Edit View Profile Scene Collection Tools Help
Scenes
Sources
Audio Mixer
Scene Transitions
Controls
Scene
Screen Capture (XSHM
Desktop Audio
Mic/Aux
Fade
Duration 300 ms
Start Streaming
Start Recording
Studio Mode
Settings
Exit
LIVE: 00:00:00
REC: 00:00:00
CPU: 2.1%, 14.06 fps
Size
Modified
14 items
11 Aug
9.4 kB
12 Aug
5.6 kB
12 Aug
0 bytes
5 Aug
4.9 kB
12 Aug
3.5 kB
12 Aug

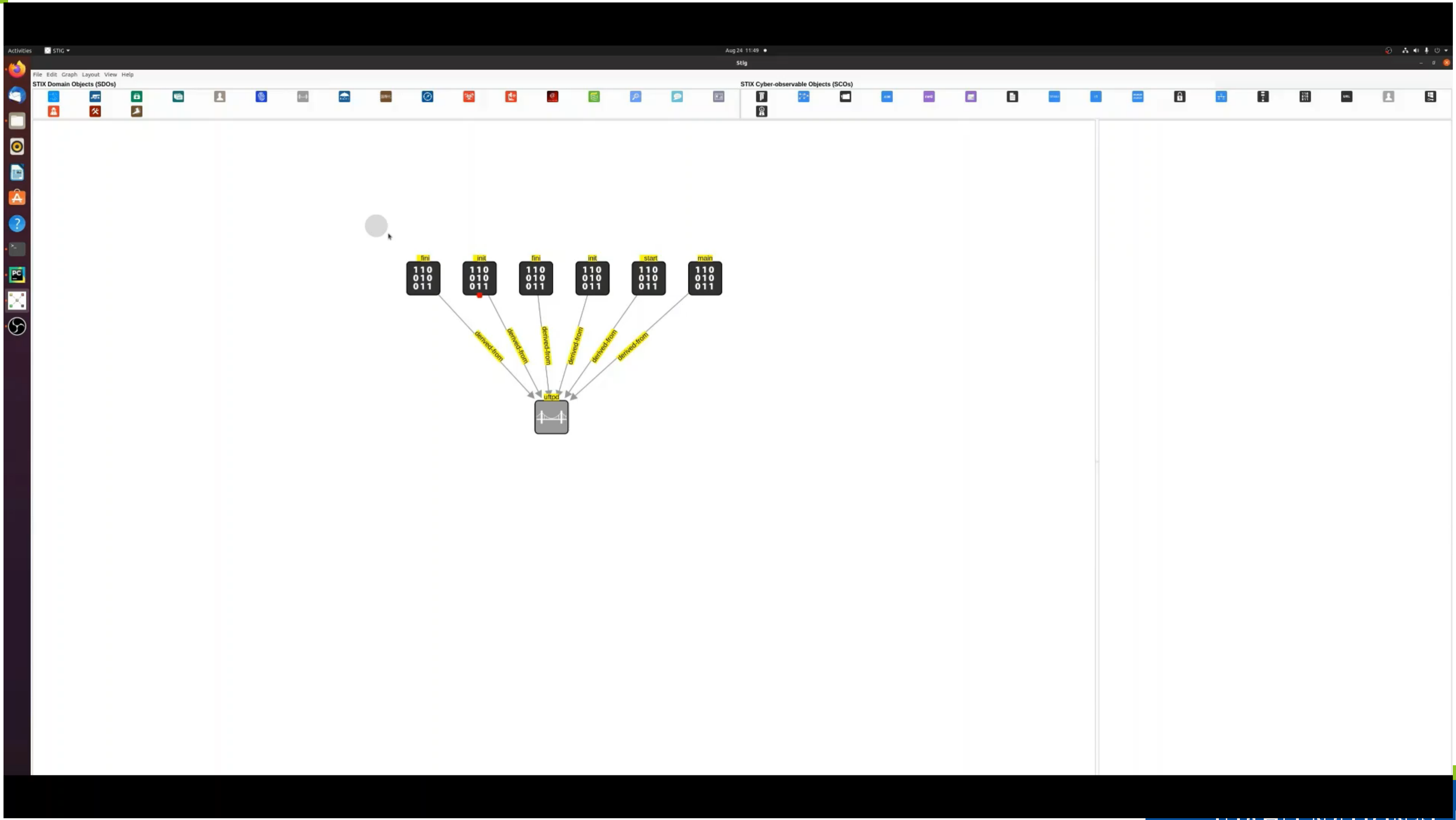
Next Steps: Binary Visualization from SBOM

- Automating vulnerabilities





Project Supporting our Efforts



Questions

Sources

- https://cyclonedx.org/
- https://oasis-open.github.io/cti-documentation
- https://spdx.dev/
- https://csrc.nist.gov/projects/software-identification-swid