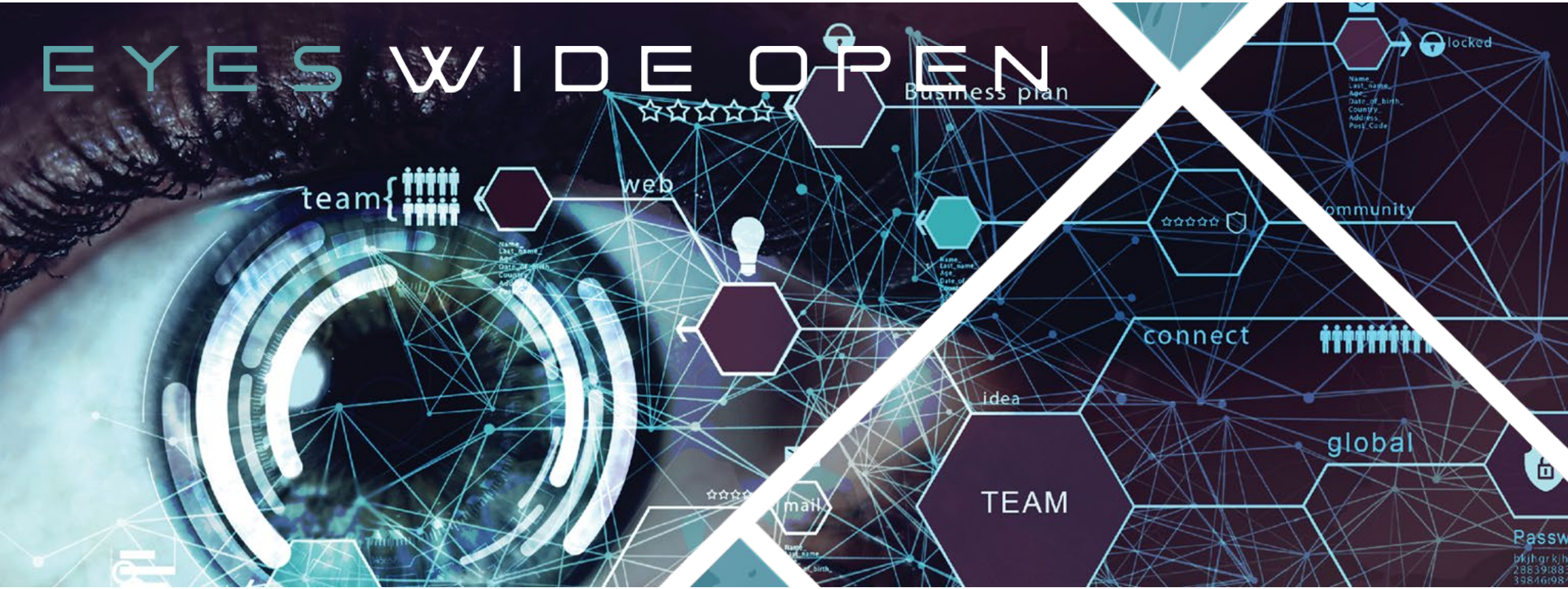
12TH ANNUAL LEADERSHIP EVENT

CYBER SECURITY SUMMIT

Security solutions through collaboration.™

TITLE SPONSOR

Island



Effective Vulnerability Management…through Relationships

- Building relationships to drive results-oriented remediation

Agenda

- About the State of Minnesota
- Overview of Vulnerability Management
- Power BI

About Neal & Nancy

Neal Dawson

- Supervisor of Threat and Vulnerability Management Team
- 20+ years state service
- 14 years with Enterprise Security Office
- Loves technology, programming, and data analytics
- Enjoys biking, hiking and watching kid's sporting events.

Nancy Skuta

- Lead Security Analyst
- 20+ years state service
- 10+ years with Enterprise Security Office
- Enjoys finding solutions to complex vulnerabilities and IT problems
- Avid traveler, hospice volunteer, and enjoys walking and hiking both on familiar and new trails

Minnesota IT Services

- Minnesota IT Services (MNIT) is the central IT organization for the State of Minnesota
  - CIO, Tarek Tomes, appointed by the governor
- MNIT delivers IT Services through a hybrid model of service delivery
- 13 Chief Business Technology Officers
  - MNIT Teams support agency-specific applications,
- Enterprise Services
  - Provides services that maximize efficiencies and reduce costs
  - services include tools like email, desktop support, and telephone services.
  - MNIT charges enterprise rates for its Enterprise Services

MINNESOTA IT SERVICES

2,400+
People employed by MNIT

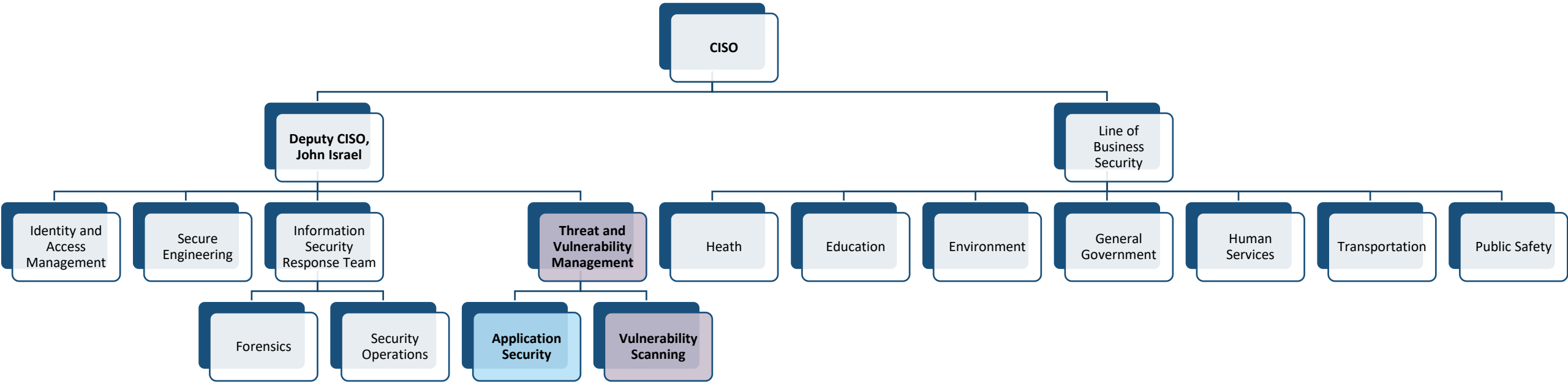
2,000+
Applications supported

540+
IT projects tracked

114+ million
Visits to state websites in 2021

Tarek Tomes, CIO

Enterprise Security

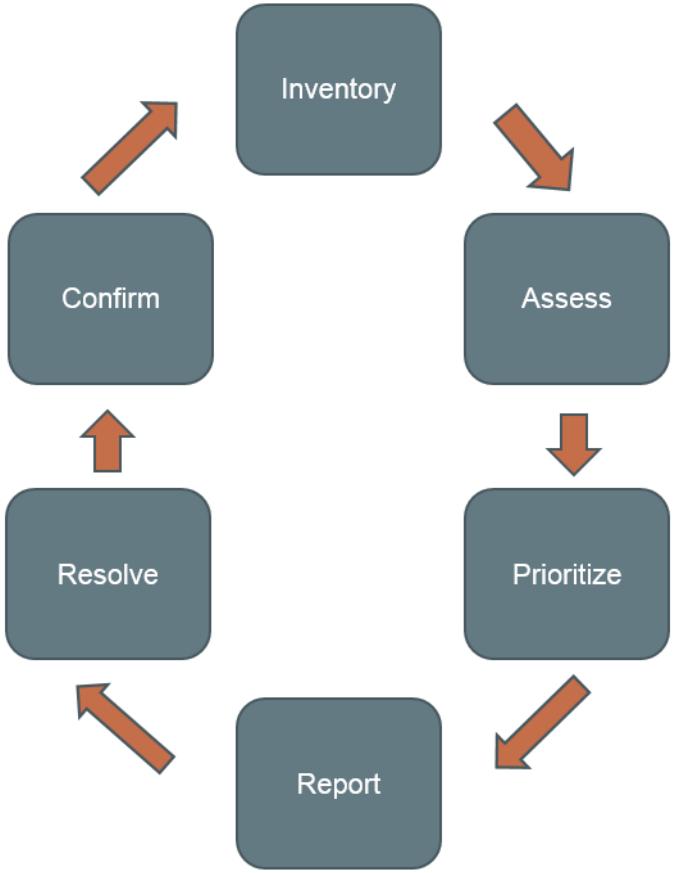


Threat and Vulnerability Management Unit (TVMU)

- Team of 10 security professionals
  - Application Security
  - Vulnerability Management Service
- Responsible for vulnerability management for the executive branch
  - Provide services for the AG and SOS offices
  - Expanding into service offering to counties
- 77 agencies, boards, and commissions
- Scan over 60,000 computing devices
- Currently Using Tenable

Vulnerability Management - What is it?

- The process to identify and resolve vulnerabilities on information systems
- Regular scans to identify vulnerabilities and resolutions
- Work with operational teams to resolve the vulnerability



Approach to Vulnerability Management

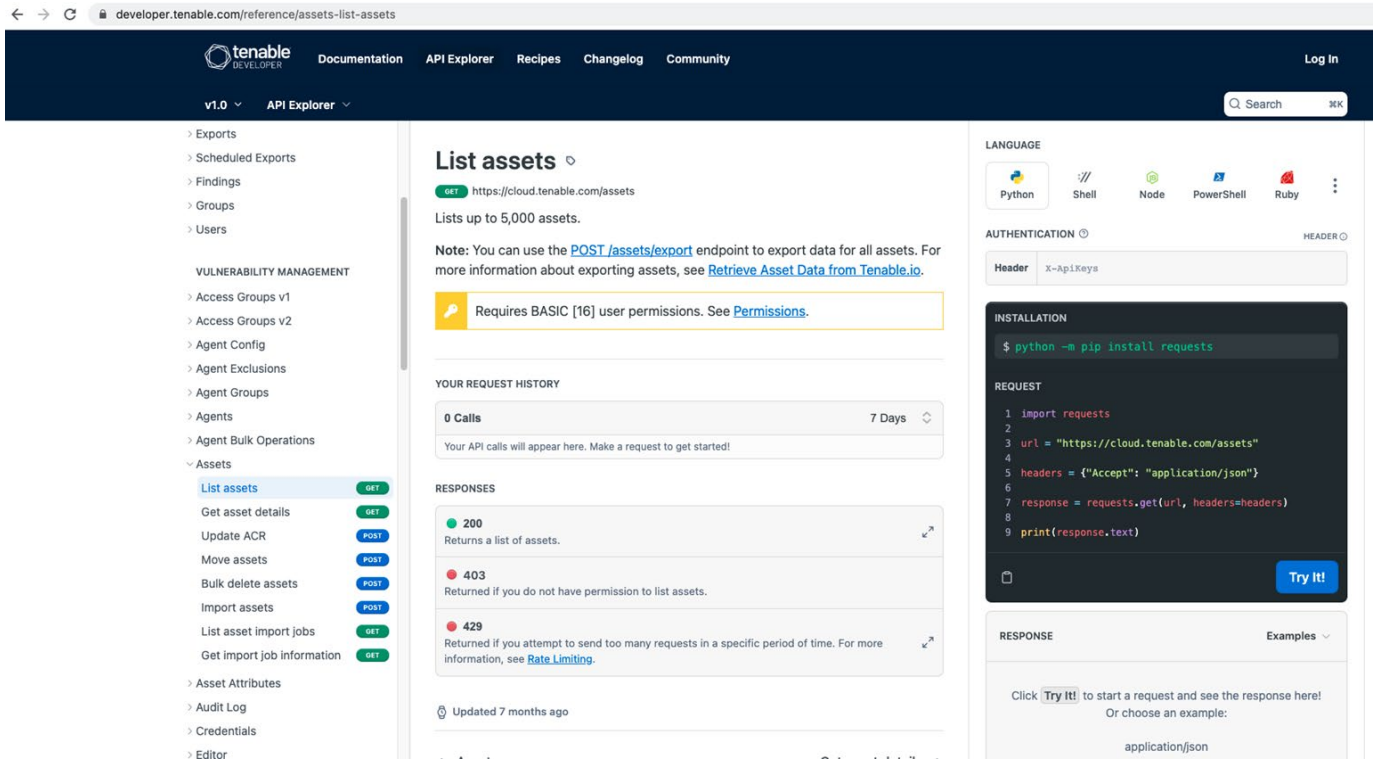
- Shifted Approach
  - No More 1000-page reports
  - Hands on / Build Relationships
    - Drove, walked, biked to Agency Locations–
      - Roseville, Apple Valley, Minneapolis
      - Met in person
      - Solved technical challenges
      - Implemented monthly meetings
- Utilizing PowerBI
- We are all part of the same team and all part of securing our State systems

Power BI

- Microsoft Reporting Platform– “Next-Level Data Visualizations”
- Very Fast
- Can handle large data sets
- Easy to drill in and filter
- Saves time and provides efficiencies

Getting Data to Power BI

- Automation of Vulnerability Management data to relational database (MS SQL Server)
- Tenable API and Python (daily updates)
- Merlin– IT data warehouse
- Power BI Service & Data Gateway



Report, Resolve, & Confirm

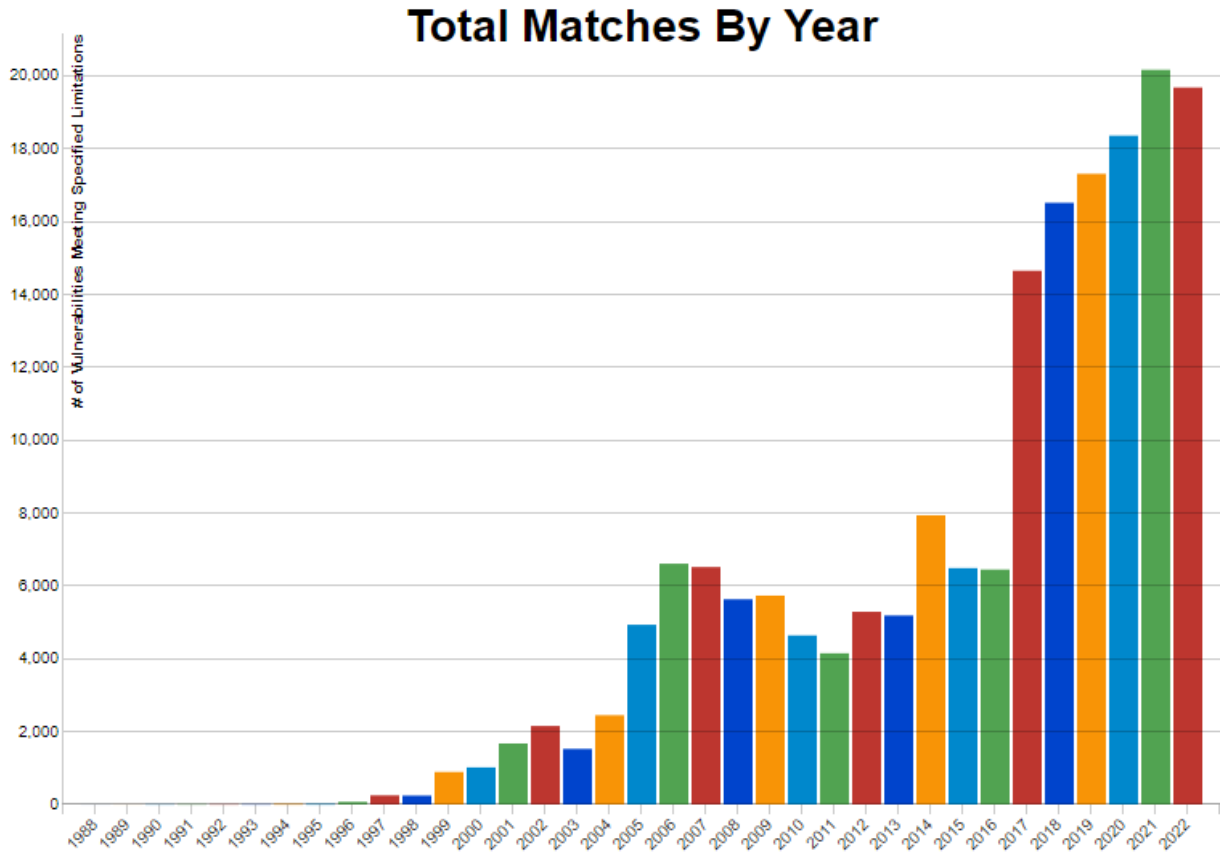
- Program revolves around these processes
  - Clear Expectations (Standards)
  - Metrics
  - Communicate the information to appropriate teams
  - Dedicated meetings with MNIT IT Teams (monthly / quarterly)
    - Currently 40 ongoing meetings
    - Enterprise teams - bi-weekly or monthly
    - Agency application teams -- quarterly
  - Identify and track vulnerabilities based on application group and severity level
  - Vulnerability information is prioritized
  - Remediation expectations clearly communicated
  - Emphasis on relationships

Vulnerabilites

- With the number of CVEs identified on an annual basis it is critical to prioritize vulnerabilities based on risk
  - https://nvd.nist.gov/vuln/search

Search Parameters:

- Results Type: Statistics
- Search Type: Search All
- CPE Name Search: false



Vulnerability Management Standard

- A set of 8 Control Requirements
- MNIT Required to comply with control requirements
  - Use Tenable VPR rating over traditional severity ratings, such as CVSS
  - Based on risk and considers current threat landscape

MNIT Standard and Mappings

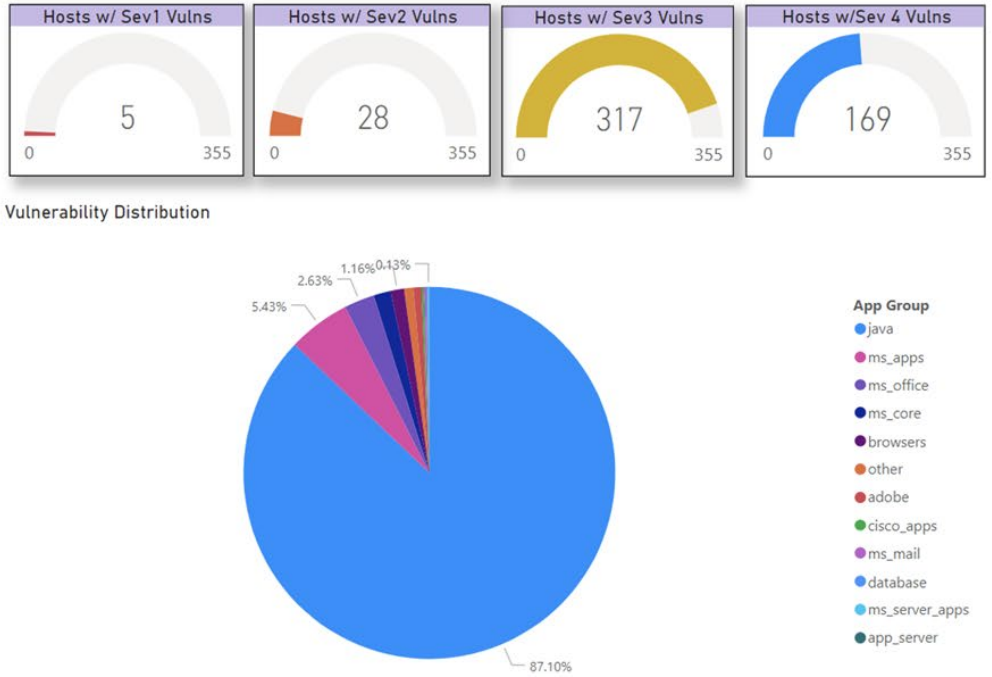
Sev Rating	Risk Rating	Remediation Expectation	Internal Scans	External Scans	Additions
1	Vulnerabilities that pose the highest risk to State systems or data and require immediate attention.	7 days	*	9.0 - 10	*Other Threat Intel or Activity specially targeted at State of MN
2	Vulnerabilities that pose great risk to State systems or data.	30 days	8.4 – 10	7.0 – 8.9	
3	Vulnerabilities that pose a moderate or indirect risk to State systems or data	90 days	6.0 – 8.3	4 – 6.9	Vulns with Critical Severity rating with VPR < 7.0
4	Vulnerabilities that only expose non-critical system information.	Next maintenance cycle	< 6.0	< 4	

- Use Tenable Vulnerability Priority Rating (VPR) over traditional severity ratings, such as CVSS
- Based on risk and considers current threat landscape
- https://mn.gov/mnit/assets/Threat%20and%20Vulnerability%20Management%20Standard_tcm38-323798.pdf

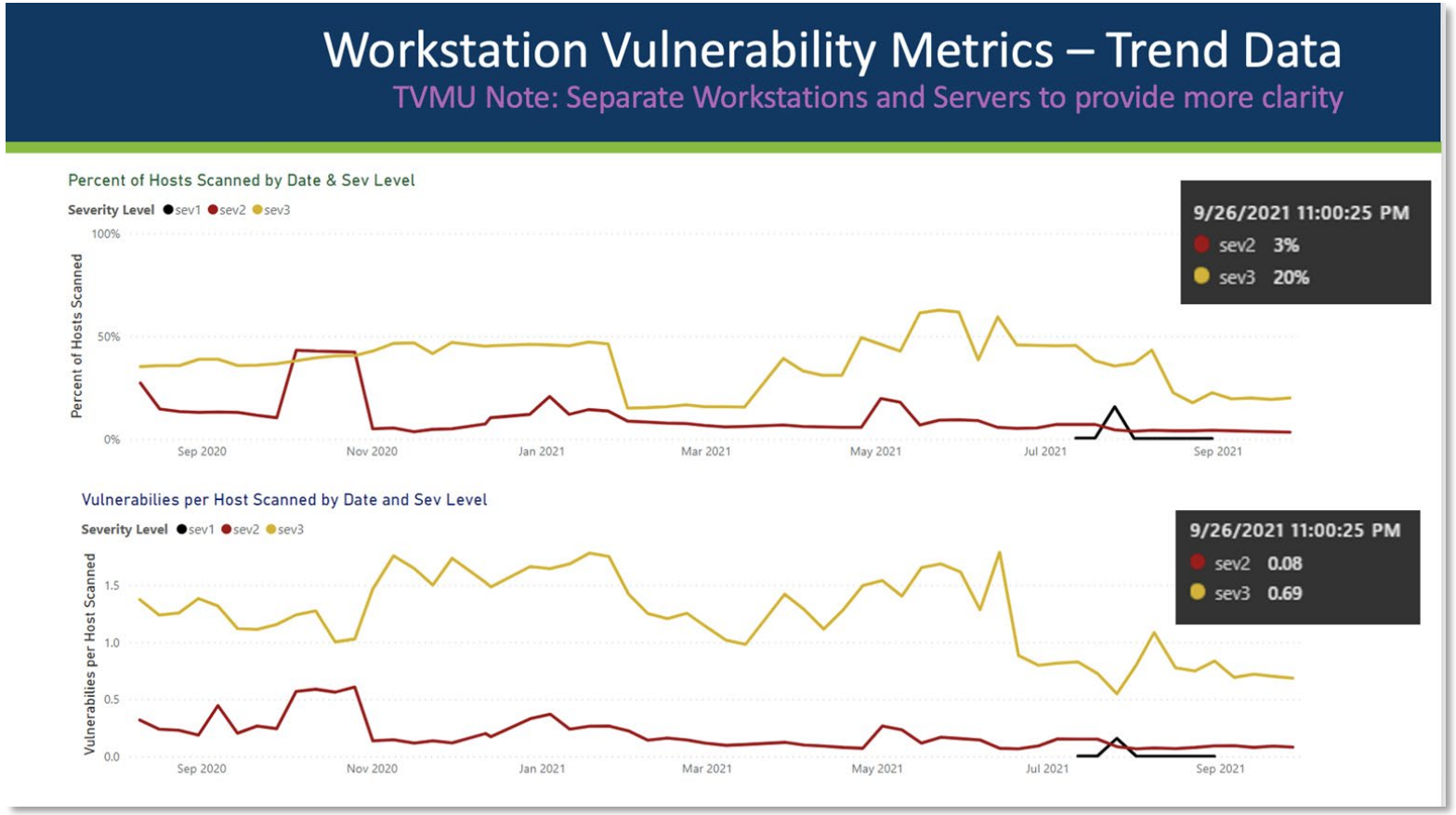
Metrics– Current Posture

Overdue Vulnerability Metrics - Workstations

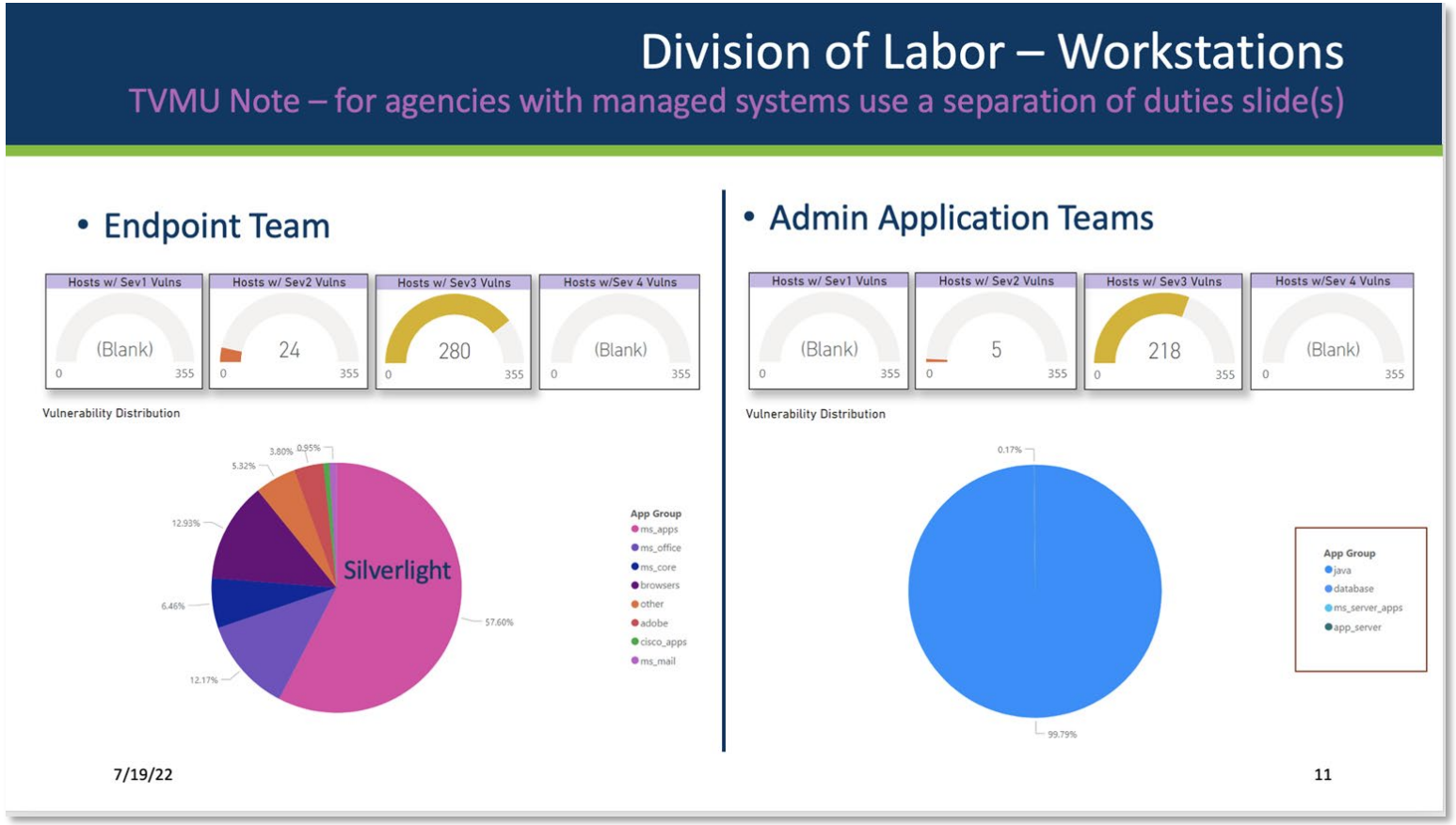
- You would choose your specific agency and make notes accordingly. Example below:
- 94% of hosts reached with authentication
- 5 workstations with a Severity 1
- 28 Hosts with one or more Severity 2 vulnerability
- The largest category is Java



Vulnerability Metrics -Trends



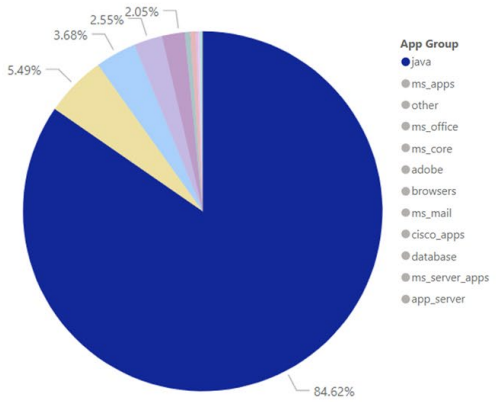
Metrics– Posture by Responsibility



Focus on Risky / Most Prevalent

Java - Workstations

- Numerous versions of Java detected
- 355 Workstations detected with Java 1.8.0_282
- 205 with Java 1.5.0_17
- 154 with Java 1.6.0_75
- Majority coming from the use of tools with Java embedded in it



Count of Hosts by Java Version

Number of Hosts

419

Total Instances

605

Agency	Hostname	OS	OS Build	System User	IP	Java Version	Path	Vendor	Last Scan Date
Administration	G02-1949882	Windows 10	21H2		192.168.1.21	1.6.0_29	D:\Windows\System32\	unknown	8/4/2022 4:04:14 AM
Administration	G02-1949882	Windows 10	21H2		192.168.1.21	1.6.0.29.11	D:\Program Files\Java\jre6\	Oracle Java	8/4/2022 4:04:14 AM
Administration	G02-1949882	Windows 10	21H2		192.168.1.21	1.6.0_05	D:\Program Files\Java\jre1.6.0_05\	unknown	8/4/2022 4:04:14 AM
Administration	G02-1949882	Windows 10	21H2		192.168.1.21	1.6.0_03	D:\Program Files\Java\jre1.6.0_03\	unknown	8/4/2022 4:04:14 AM
Administration	G02-1949882	Windows 10	21H2		192.168.1.21	1.6.0_00	D:\Program Files\Java\jre1.6.0\	unknown	8/4/2022 4:04:14 AM
Administration	G02-1949060	Windows 10	21H2	Anbar Ahmed	10.77.121.184	1.8.0.291.09	C:\sqldeveloper\jdk\	Oracle Java	8/4/2022 12:25:23 A
Administration	G02-1839473	Windows 10	21H2	Kari Suchy	10.77.253.171	1.11.54.26	C:\Program Files\Tableau\Tableau Reader 2022.1\bin\	Azul Zulu Java	8/3/2022 6:38:39 PM
Administration	G02-1847429	Windows 10	21H2	Antonette Johnson	10.77.253.157	1.11.54.26	C:\Program Files\Tableau\Tableau Reader 2022.1\bin\	Azul Zulu Java	8/3/2022 4:28:19 PM
Administration	G02-1647707	Windows 10	21H2		10.77.120.228	1.11.52.14	C:\Program Files\Tableau\Tableau Reader 2021.4\bin\	Azul Zulu Java	8/3/2022 8:25:32 PM
Administration	G02-1852979	Windows 10	21H2		10.77.121.213	1.11.52.14	C:\Program Files\Tableau\Tableau Reader 2021.4\bin\	Azul Zulu Java	
Administration	G02-1874614	Windows 10	21H2		10.0.0.42	1.11.52.14	C:\Program Files\Tableau\Tableau Reader 2021.4\bin\	Azul Zulu Java	8/4/2022 6:52:07 AM
Administration	G02-1878388	Windows 10	21H2		10.77.121.143	1.11.52.14	C:\Program Files\Tableau\Tableau Reader 2021.4\bin\	Azul Zulu Java	8/4/2022 1:25:11 AM

Highlights of Key Vulnerabilities

Application Severity 2 Vulns

- Highlight application related vulnerabilities
  - Include appropriate details, example below:
  - GXX-1874288, Kevin Jones
  - C:\junk\ase6\jdk1.7.0_45x86\jre\bin\java.exe
    - Installed version: 1.7.0_45
  - Can the java be removed from the Junk folder?

- Highlight application related vulnerabilities
  - Include appropriate details, example below:
  - GXX-AVIGILON1, Sue Smith
  - Path: C:\Program Files (x86)\MegaRAID Storage Manager\JRE\bin\java.exe
    - Installed version: 1.6.0_24
    - MegaRAID Storage Manager v12.08.03.0300 [version 12.08.03.0300] [installed on 2013/02/01]
  - Can the MegaRAID Storage Manager be updated?

End of Life Technology

Application EOL & Life Cycle

- ArcGIS 10.2 - July 01, 2019
- ArcGIS 10.3 - December 01, 2020
- ArcGIS 10.8. 1 - March 01, 2026
- PostgreSQL 9.6 - November 11th, 2021
- PostgreSQL 10.12 - November 10, 2022
- PostgreSQL 14.0 - November 12, 2026
- Tableau versions 2019.3.x, 2019.4.x, 2020.1.x, 2020.2.x, and 2020.3.

7/19/22

https://www.cisecurity.org/blog/end-of-support-report-list/

Server 2012
T -15 Months

- Identify the percent of Windows Servers in the Agency that are on a Server 2012 platform
- It is not to early to start planning for a migration off of 2012!
- Windows Server 2012 and 2012 R2 - EOL 10/10/2023

8/5/2022

30

Progress / Encouragement

Some Good News

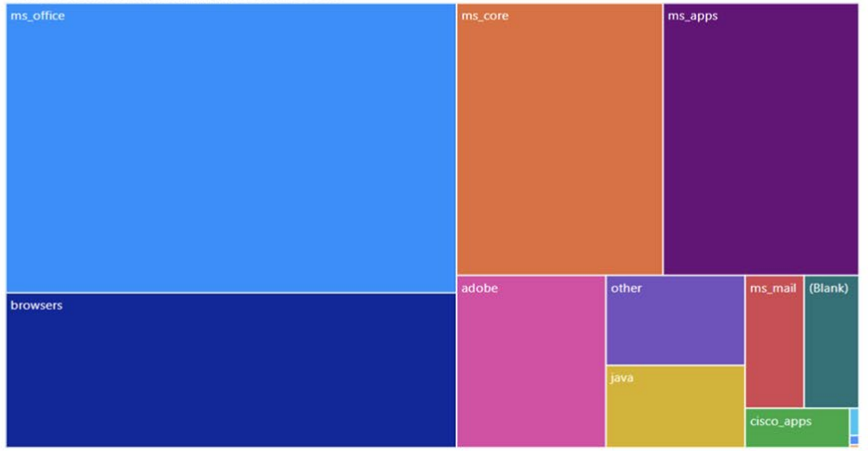
- Add in information on the number of Vulnerabilites remediated for the Agency/Group you are working with

Number of Vulnerabilites Mitigated in Accordance with the Standard in the Last 30 Days

306019

vulns

Vulnerabilities Mitigated by Application Group



7/19/22

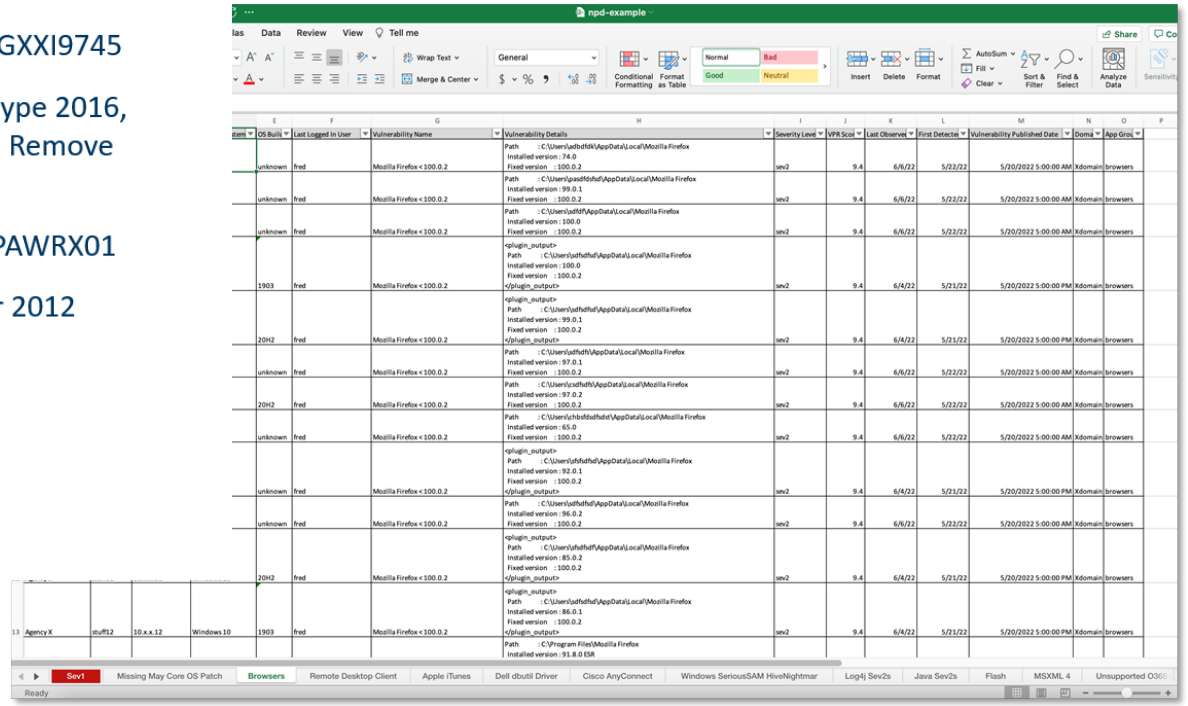
Recap & Support Information

Recap & Action Items

TVMU Note – Try to be specific on some things the team can do to fix vulnerabilities.

- Workstations
  - ❑ Update or Remove Adobe After Effects on BXX-321987
  - ❑ Remove Google Chrome from the user profile on BXX-123456
  - ❑ Remove ArcGIS 10.3 on CARP-1
  - ❑ Update SQLdeveloper - 16 Workstations
  - ❑ Ensure the SCCM Client on GXX-18903 is in good working order and ensure delinquent patches are applied
  - ❑ Update ArcGIS Pro - 50 workstations

- Servers
  - ❑ Resolve the WebLogic Sev 1 on GXXI9745
  - ❑ Evaluate GXXPAWBRDS01 for Skype 2016, Outlook 2016, and Adobe Flash, Remove unneeded software
  - ❑ Update Apache Tomcat on GXXPAWRX01
  - ❑ Continue to work toward Server 2012 retirements



8/5/2022

TVMU Goal & Outcomes

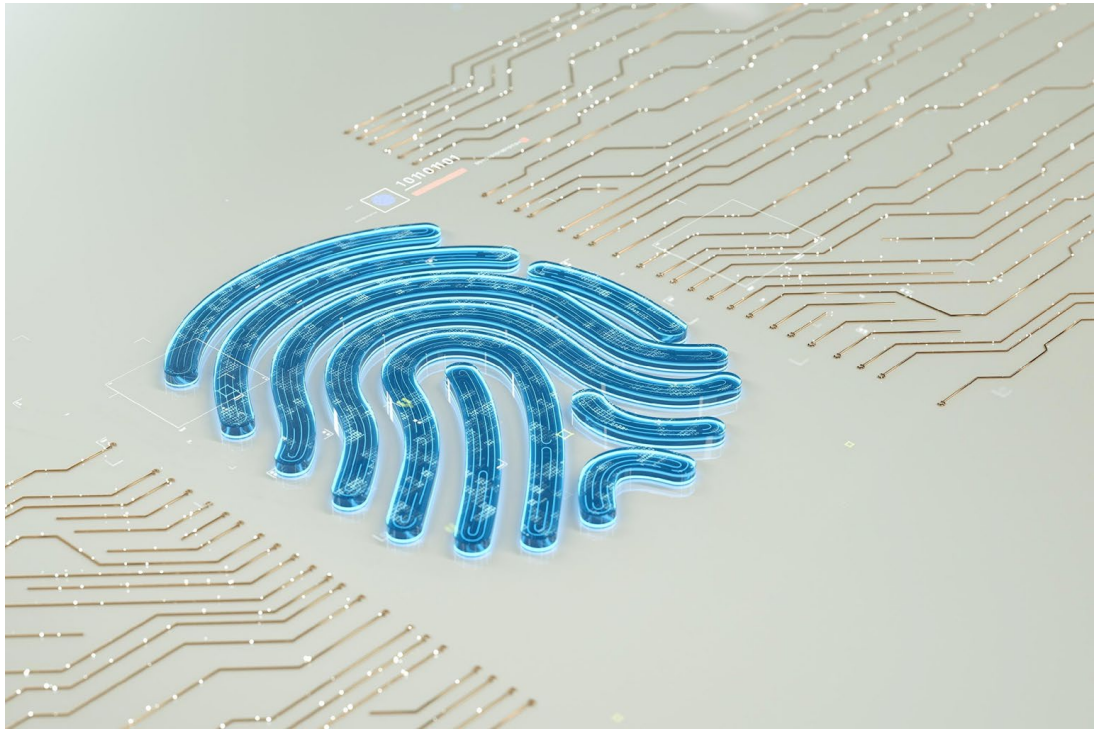
- Goal:
  - To work collaboratively with MNIT teams to secure our systems!
- Outcomes:
  - Remediation timeframes have improved
  - Reduction in vulnerabilities and risk exposure
  - Teams reach out to us

Power BI

How the State of Minnesota security team utilizes Power BI for Vulnerability Management

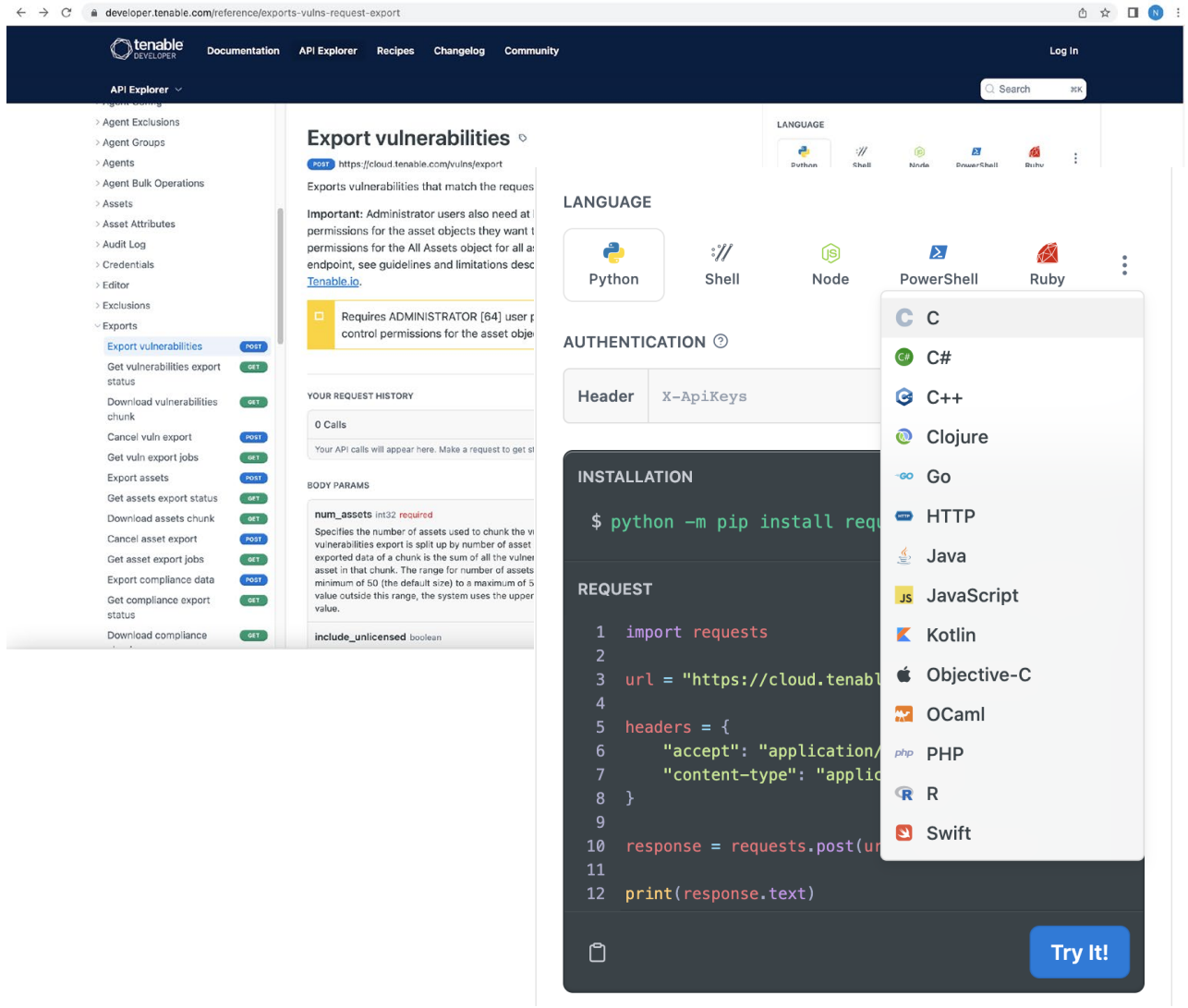
Power BI - Uncovered

- API
- Database & Gateway
- Extract, Transform, Load (ETL)
- Power BI Environment



Using Scanning Tools API

- Web API
- Tenable utilizes REST API endpoints.
- Programmatically interact with the platform
  - Can execute scans
  - Create scan polices
  - Export data
- Simple to use (after learning curve)
- Multiple Languages and Scripts



Vendor APIs

- Most scanning vendors have APIs
- Actually, most tool tools have APIs

RAPID7

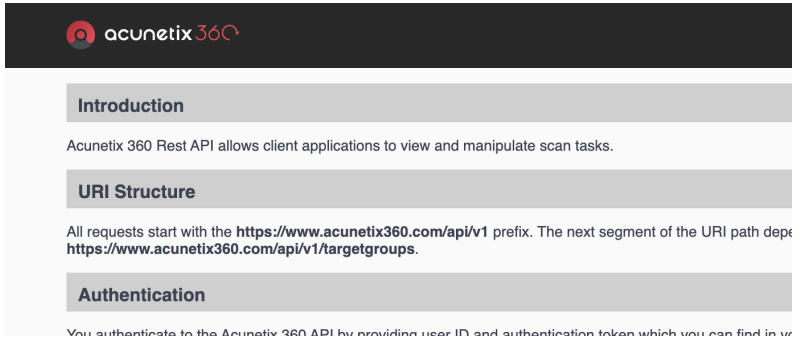
Qualys.

CROWDSTRIKE

Qualys API (VM, PC)

User Guide
Version 10.20.4

October 12, 2022


acunetix 360

Introduction

Acunetix 360 Rest API allows client applications to view and manipulate scan tasks.

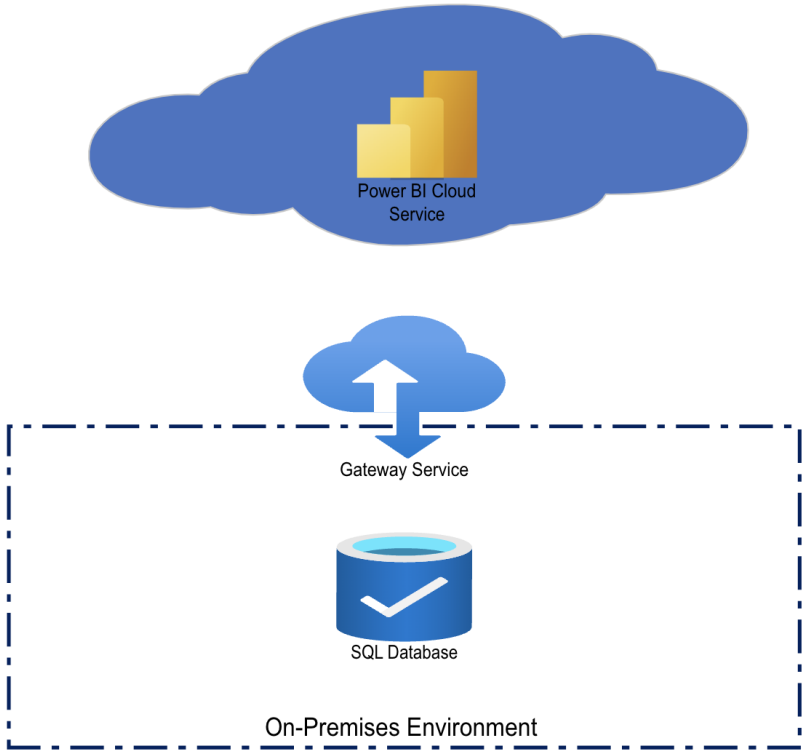
URI Structure

All requests start with the **https://www.acunetix360.com/api/v1** prefix. The next segment of the URI path depe
https://www.acunetix360.com/api/v1/targetgroups.

Authentication

SQL Server Database & Gateway

- SQL Database for Reporting Data
- Database design for reporting– **This is important**
  - Understand key data elements needed for reporting
- Power BI Gateway
  - Bridge between on-premises database and Microsoft cloud service



Using API to extract data

```python
def getAssets(headers):
    try:
        counter = 1
        i = 1
        last_week = int(time.time()) - 604800
        payload = {
            "chunk_size": 5000,
            "filters": {
                "last_authenticated_scan_time": last_week
            }
        }
        url = "https://cloud.tenable.com/assets/export"
        response = requests.request("POST", url, json=payload, headers=headers)
        print(response.json())
        export_id = response.json()['export_uuid']
        #check status
        url = "https://cloud.tenable.com/assets/export/" + export_id + "/status"
        response = requests.request("GET", url, headers=headers)
        print(response.json())
        chunk_completed=[]
        checkjob = True
        while checkjob:
            if response.json()['status'] == 'FINISHED':
                checkjob = False
               # url = "https://cloud.tenable.com/assets/export/" + export_id + "/chunks/1"
                #url = "https://cloud.tenable.com/assets/export/export_uuid/chunks/chun"
                for chunck in response.json()['chunks_available']:
                   print(str(chunck))
                   if chunck not in chunk_completed:
                       url2 = "https://cloud.tenable.com/assets/export/" + export_id + "/chunks/" + str(chunck)
                       response2 = requests.request("GET", url2, headers=headers)
                       chunk_completed.append(chunck)
                       for asset in response2.json():
                           print(asset)
                           id = [asset['id']]
```

```python
# imports pyTenable
from tenable.io import TenableIO
import requests
import time

def getTestVulns(api_key,api_secret):
    last_week = int(time.time()) - 604800
    tio = TenableIO(api_key,api_secret)
    for vuln in tio.exports.vulns(tags=[('agency','BAH')],severity=['critical','high'],last_found = last_week):
        print(vuln)
        uuid = vuln['asset']['uuid']
        hostname=vuln['asset']['hostname']
```

PyTenable

https://pytenable.readthedocs.io/en/stable/

Search

- Tenable.io
- Tenable.sc
- Tenable.ot
- Product Downloads
- Tenable.ad
- Nessus
- Reports
- Cross-Package Tooling
- Common Components
- Errors
- Testing the Library

Welcome to pyTenable's documentation!

build success | pypi v1.4.8 | python 3.6 | 3.7 | 3.8 | 3.9 | 3.10 | downloads 58k/month | license MIT

pyTenable is intended to be a pythonic interface into the Tenable application APIs. Further by providing a common interface and a common structure between all of the various applications, we can ease the transition from the vastly different APIs between some of the products.

- Issue Tracker: https://github.com/tenable/pyTenable/issues
- Github Repository: https://github.com/tenable/pyTenable

Note

pyTenable version 1.2 is the first release to drop support for Python versions less than 3.6. If you are still using any of these deprecated python versions the 1.1 release will still work, however is no longer being actively developed or maintained.

Installation

To install the most recent published version to pypi, its simply a matter of installing via pip:

```
pip install pytenable
```

If your looking for bleeding-edge, then feel free to install directly from the github repository like so:

```
pip install git+git://github.com/tenable/pytenable.git#egg=pytenable
```

Getting Started

Lets assume that we want to get the list of scans that have been run on our Tenable.io application. Performing this action is as simple as the following:

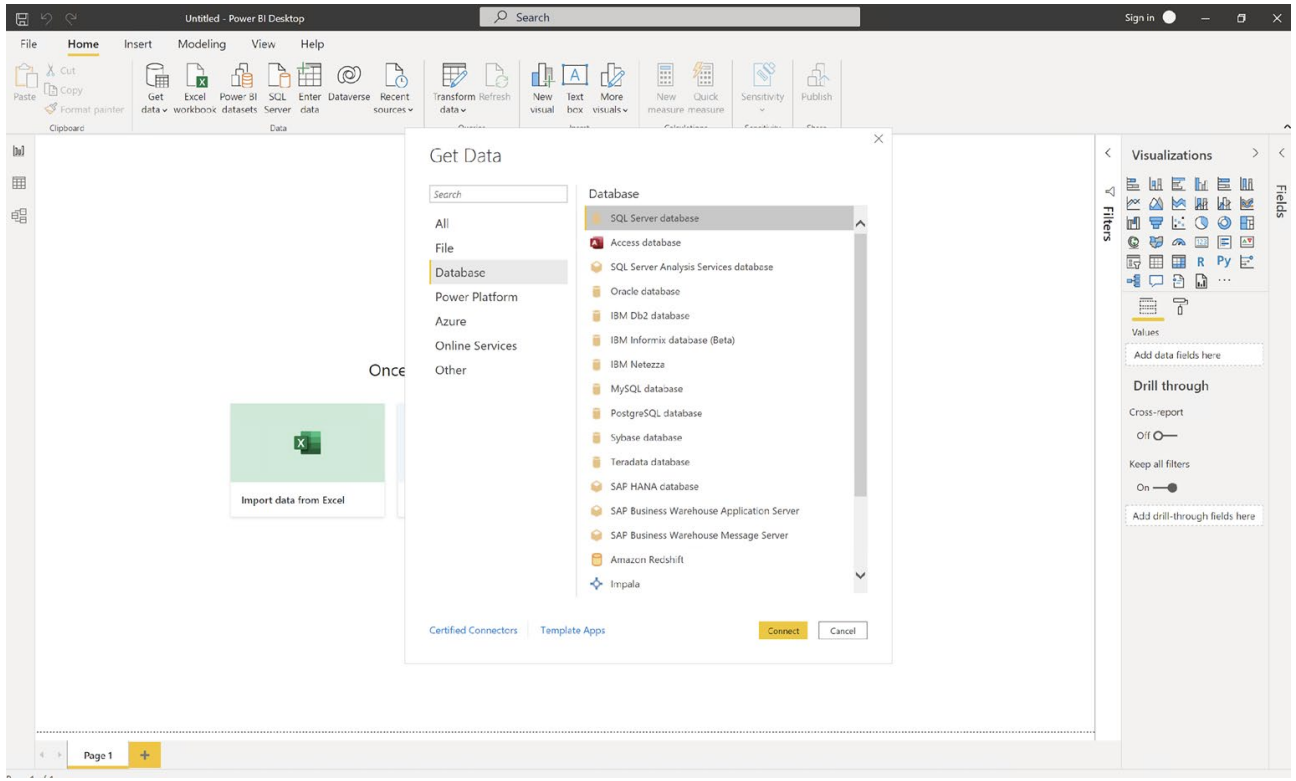
Automating the ETL programs

- Build Programs
- Test, Test, Test
- Set up Scheduled Jobs
- Log and monitor

```
[nzeek@mysuperserver ~]$ crontab -l
0 21 * * *  /data/tye/scripts/runEtlJobs.sh
0 9  * * *  /data/tye/scripts/runGetPlugins.sh
0 11 * * *  /data/tye/scripts/dostuff.sh
0 14 * * *  /data/tye/scripts/runCloudInventory.sh
0 8  * * 1  /data/tye/scripts/dothings.sh
0 11 2-30/2 * * /data/tye/getNvdCisa.sh
[nzeek@mysuperserver ~]$
```

Power BI Report Development

- Using Power BI Desktop
- Connecting to SQL / Data Source
- Dataset Modes
  - 3 Types (Import, Direct Query, Composite
- Import Mode
  - Very Fast
  - Compressed
  - Some limitations



Key Metrics & Reports

- Scan Coverage
- Percent of hosts with overdue vulnerabilities
  - Sev1, Sev2, Sev3
- Average Vulnerability Age

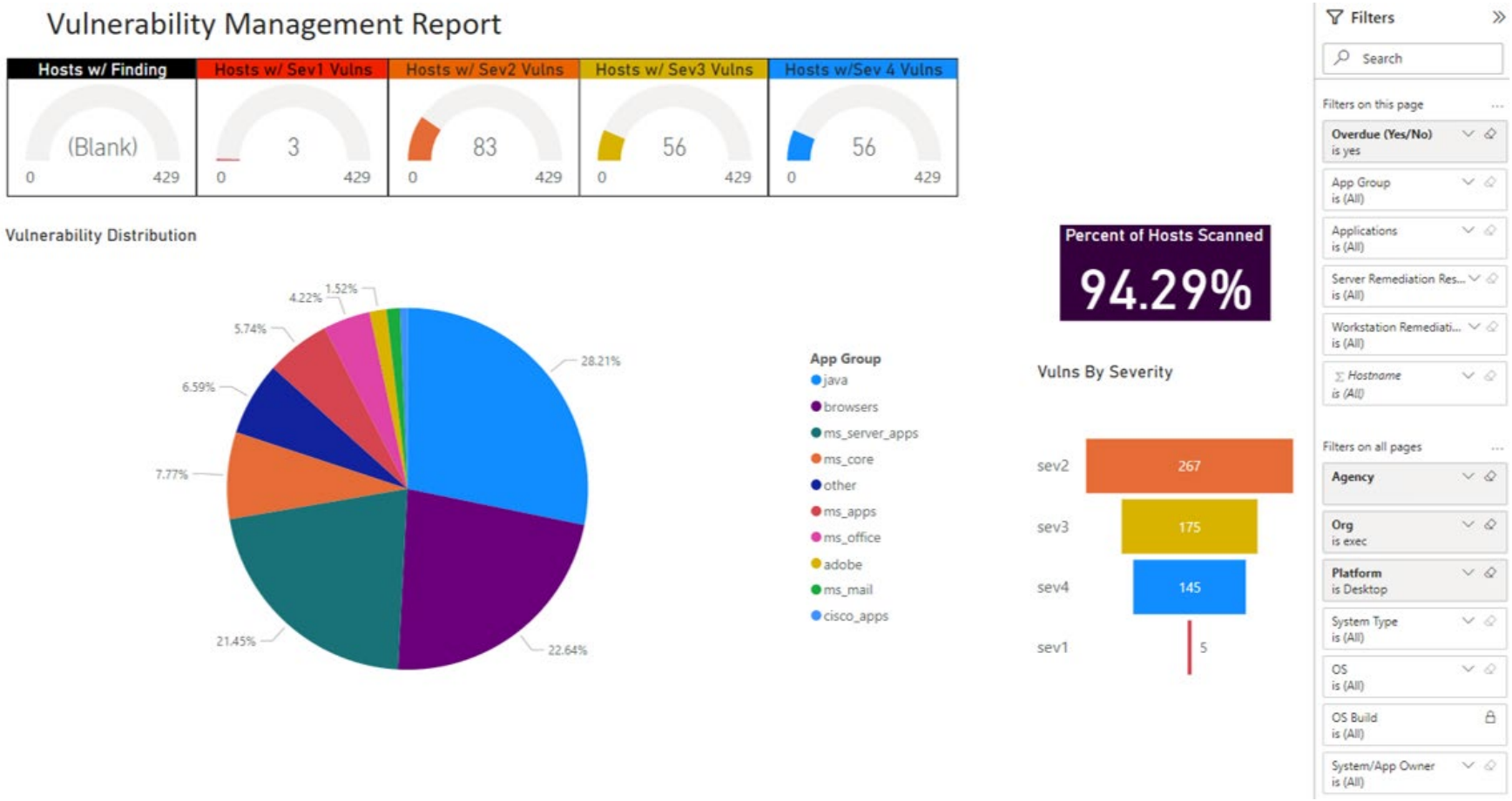
Executive Summary

- Limited filtering
- Ability to drill into weeds (data details)
- Visuals and Trends

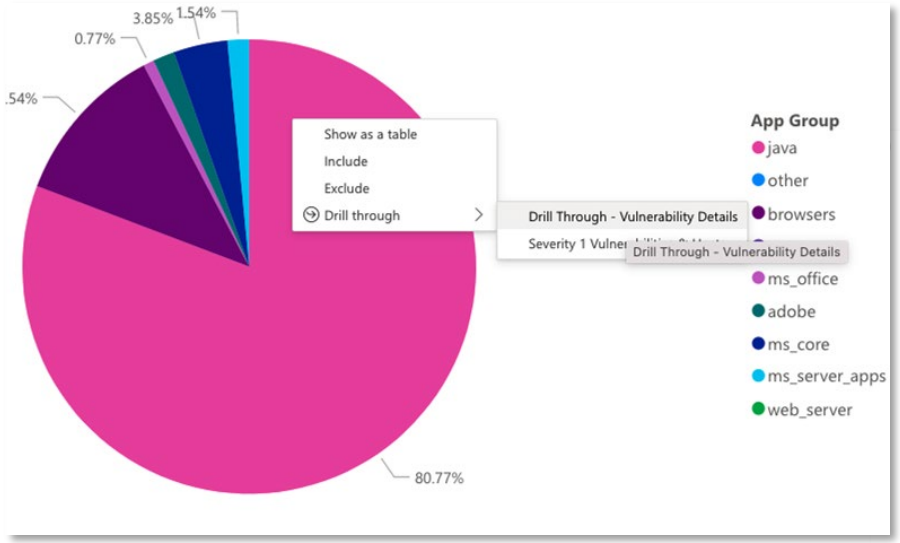
Technical Report

- Similar to Executive
- More filtering capabilities
- Drill-Thru reports more detailed fields

Executive & Technical Reports



Drill-Thru Capabilities

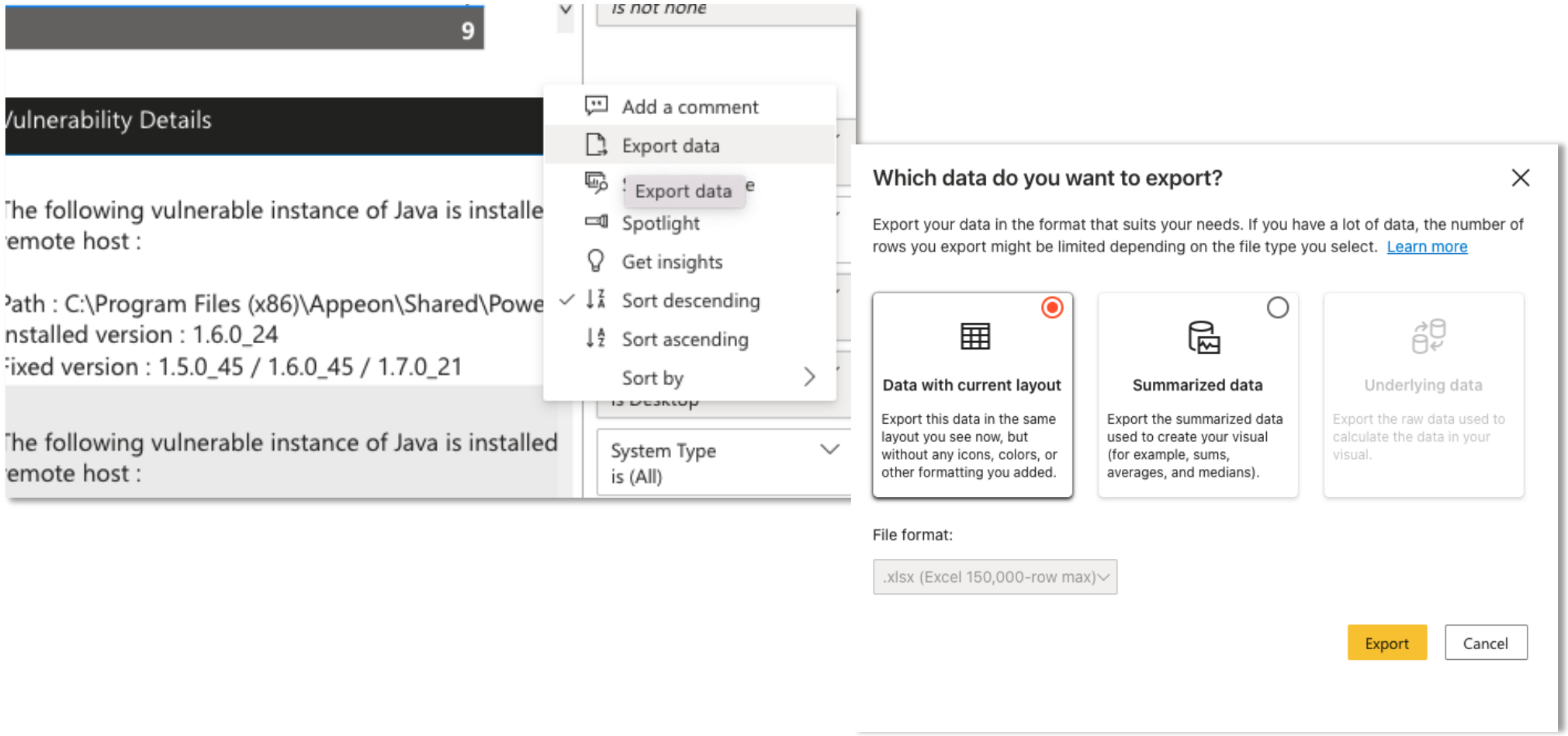


Vulnerability Details - Drill

Application Group	Plugin ID	Vulnerability Name	Severity Level	VPR Score	Number of Hosts with this Vulnerability
java	65995	Oracle Java SE Multiple Vulnerabilities (April 2013 CPU)	sev2	9.70	9
java	57959	Oracle Java SE Multiple Vulnerabilities (February 2012 CPU)	sev2	9.90	9
java	64454	Oracle Java SE Multiple Vulnerabilities (February 2013 CPU)	sev2	9.30	9
java	54997	Oracle Java SE Multiple Vulnerabilities (June 2011 CPU)	sev2	9.10	9
java	59462	Oracle Java SE Multiple Vulnerabilities (June 2012 CPU)	sev2	9.90	9
java	66932	Oracle Java SE Multiple Vulnerabilities (June 2013 CPU)	sev2	9.70	9
java	56566	Oracle Java SE Multiple Vulnerabilities (October 2011 CPU) (BEAST)	sev2	9.80	9
java	62593	Oracle Java SE Multiple Vulnerabilities (October 2012 CPU)	sev2	9.80	9
java	78481	Oracle Java SE Multiple Vulnerabilities (October 2014 CPU)	sev2	9.90	9
java	65050	Oracle Java JDK / JRE 6 < Update 43 Remote Code Execution (Windows)	sev2	9.80	7
Total					**9**

Agency	Host Name	IP	Operating System	OS Build	Last Logged In User	Vulnerability Name	Plugin ID	Vulnerability Details
Labor and Industry			Windows 10	21H2		Oracle Java SE Multiple Vulnerabilities (April 2013 CPU)	65995	The following vulnerable instance of Java is installed remote host : Path : C:\Program Files (x86)\Appeon\Shared\PowerI Installed version : 1.6.0_24 Fixed version : 1.5.0_45 / 1.6.0_45 / 1.7.0_21
Labor and Industry			Windows 10	21H2		Oracle Java SE Multiple Vulnerabilities (April 2013 CPU)	65995	The following vulnerable instance of Java is installed remote host : Path : C:\ORA11gw\Home\jdk\jre\bin\java.exe Installed version : 1.5.0_17 Fixed version : 1.5.0_45 / 1.6.0_45 / 1.7.0_21
Labor and Industry			Windows 10	21H2		Oracle Java SE Multiple Vulnerabilities (April 2013 CPU)	65995	The following vulnerable instance of Java is installed remote host :

Exporting Data



Questions

More than happy to spend more time sharing our approach to vulnerability management and use of Power BI ☺

Nancy.Skuta@state.mn.us
Neal.Dawson@state.mn.us