Assets Risk Analysis & Triaging

Information Security and Management
G1 Slot

Abhisar Shukla	18BCE0110
Anshul Tripathi	18BCE0148
Yashaswi Shivank	18BCE0162

Overview

1. Abstract
2. Introduction
3. Survey
4. Proposed
5. Result and Conclusion
6. References

Abstract

In today’s time, vulnerability management is almost equal to *risk prioritisation* because:

- Skillset and time resources are limited in every organisation.
- The change in the environment is very fast and too frequently.
- An exponential increase in the attack surface is seen, which can only be handled by prioritizing the risks firse.
- According to the 80/20 rule — 20% of vulnerabilities bring 80% of risk.

Introduction

- Most businesses these days, big or small face more virtual problems than physical ones. The problems are related to cybersecurity.
- Risk analysis helps analyze the business's current situation. It also helps to identify, protect and manage systems, information data and resources.
- Most of the times, many vulnerabilities are found in the process of risk analysis. Many of which are at the same level of severity, they are clustered.
- A method is needed to be developed to prioritize the tasks so that the business can focus on risk management using limited resources.
- Risk analysis and risk prioritization work in tandem to provide a sustainable way for effectively targeting and resolving vulnerabilities.

Risk Analysis

- Risk analysis starts with identifying and defining all the valuable resources, such as servers, workstations, operation critical softwares, network modules, etc.
- Identifying the user of the system, with information on user location, privileges and level of access.
- Threat identification and vulnerability identification steps catalog all the errors and missteps that can result in having negative effect on the business.
- Vulnerability identification step includes identifying and cataloging all the softwares and services that are vulnerable and could end up facilitating a security breach.
- In this project we will use Nessus tools to Index all the devices connected and detect vulnerable softwares and services.

Risk Prioritization

- Risk analysis lists out all the potential risks or vulnerabilities along with their severity.
- When many risks are clustered at the same level, prioritization of them is very important.
- Risk prioritization is performed by the assigned employees of a particular organisation, provided they have all the details related to assets and the vulnerabilities present in them.
- Apart from the asset and vulnerability details, our software allows the users to perform triaging.
- The process of triage is understanding the types of vulnerabilities that should be the most worrying

Risk Prioritization

- Different types of vulnerabilities presents different types of risks. If an exploit isn’t available, that certainly lowers the risk. The method to deal with risks should also change with changing risks, which is very frequent.
- Some may enable attackers to execute denial of service attacks, some may result in escalated privileges to remote attackers, while others could allow the attacker to read or modify data.
- The above mentioned scenarios are the technical impacts of a vulnerability, and are important from a security standpoint, therefore these details will also be provided by our software along with the existing solution.

Design

We are planning to create a simple interactive and user friendly interface for IT asset management.

Technologies to be used:

django

Nessus vulnerability scanner

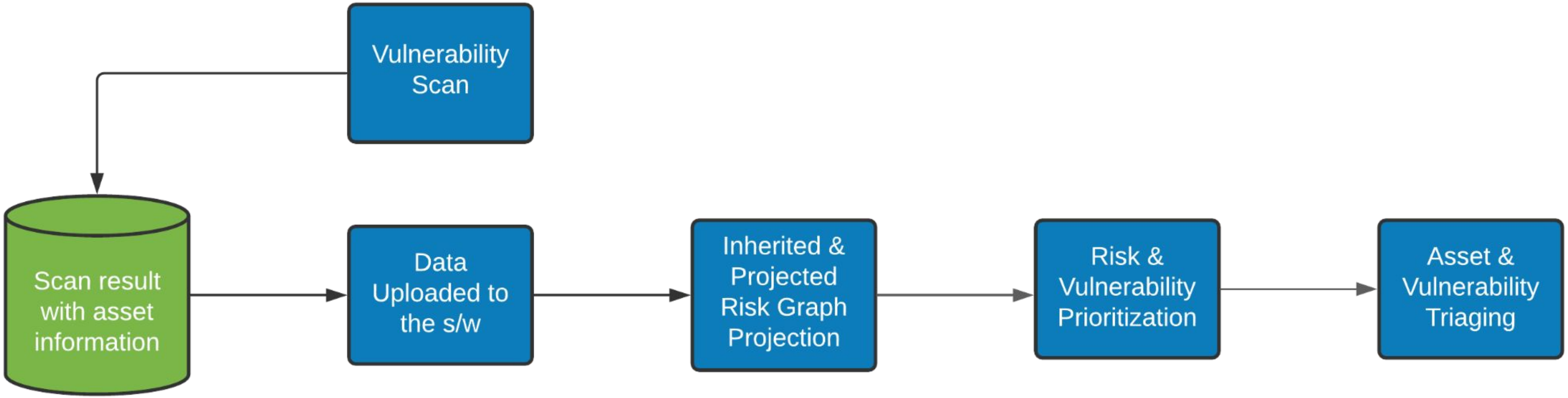
- The system is going to be built on the django web framework for developing the web application.

Design (Cont.)

Functionalities

- Assigning significance on per asset basis.
- Assessing severity on per vulnerability basis.
- Adjusting both factors at asset & vulnerability relationship level.
- Community analytics provides insights as suggested risk.
- Uploading the nessus scanned csv format output directly without any additional changes.

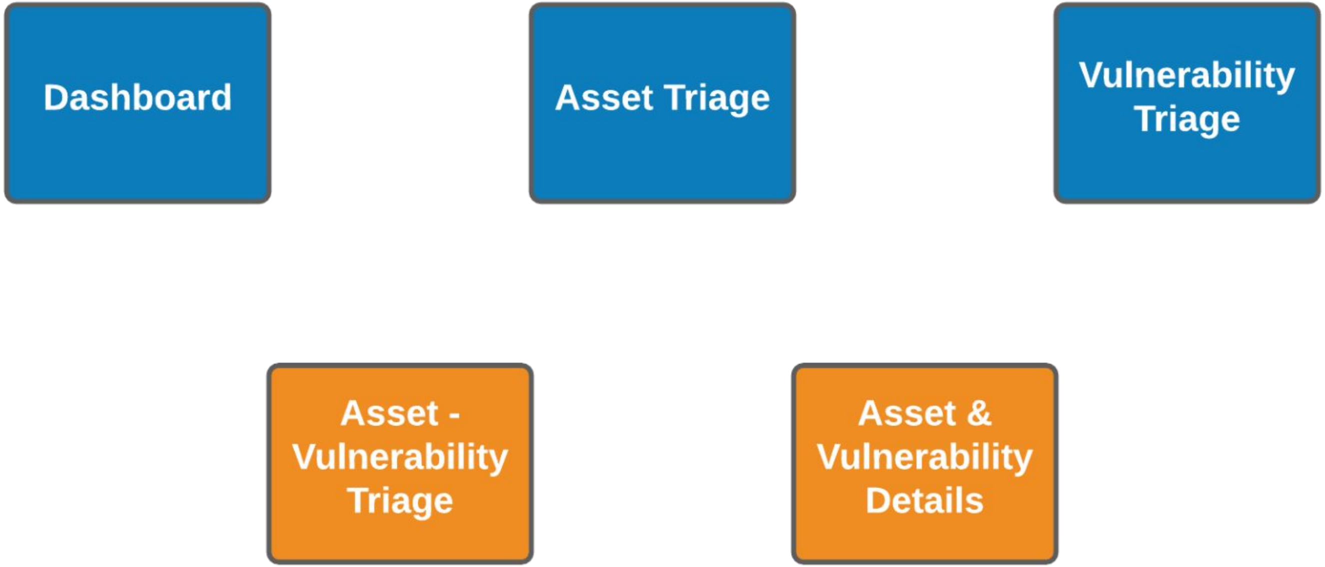
Flow of the application


Vulnerability Scan
Scan result with asset information
Data Uploaded to the s/w
Inherited & Projected Risk Graph Projection
Risk & Vulnerability Prioritization
Asset & Vulnerability Triaging

Flowchart Description

- The scan results from nessus scanner is exported as csv file and uploaded to the our developed application.
- The data contains several information related to assets which includes vulnerabilities, CVSS Score, etc.
- The inherent risk is calculated and is represented as “Projected Risk” which is also visualised as a graph for better understanding.
- The risk is calculated taking into account all the important attributes and the user is given the access to:
    - Triage the assets or vulnerabilities
    - Give priorities to risks
    - Get solution for a particular vulnerability

Modules



Dashboard

Asset Triage

Vulnerability Triage

Asset - Vulnerability Triage

Asset & Vulnerability Details

Dashboard

- This module provide information related to most important and most vulnerable assets of the company.
- According to the 80-20 rule , 20% of vulnerabilities bring 80% of risk therefore it very important to identify which assets have more chances of being compromised and which assets are the most crucial ones with large number of dependencies.
- This section also provides a graphical representation projected risks and Inherent risk which is the amount of risk that exists in the absence of controls.

Asset Triage

- This section provides a lookup on the basis of assets for different IP address and hostname
- Severity is assigned on per asset basis with various parameters.
- Parameters include criticality , Accessibility , mRisk and percentage to overall risk.
- Asset can be triaged on the basis of these parameters as critical , Moderate or Trivial assets.

Vulnerability Triage

- In this section severity is assessed on per vulnerability basis.
- Parameters considered in this section include severity (Inherited vs Projected) , Common Vulnerability Scoring System (CVSS) , mRisk and percentage to overall risk.
- Assessed severities can be remarked as critical , high , medium , low , informational , False Positive and Risk Accepted.

Asset – Vulnerability Triage

- In this section both factors at asset & vulnerability relationship level can be adjusted.
- This section has various filters that enhances the reliability of the system.
- A specific range for CVSS and mRisk can be provided which helps in the risk prioritization and decision making process.
- Based on this unified and standardized data it can be decided which vulnerabilities and ties need remedification.

-
  ```
  severityvalue * (accessibilityvalue + asset_criticalityvalue)
  ```

Asset and Vulnerability Details

- For any of the vulnerability , Application searches the NVD(National Vulnerability Database) for any known unfixed vulnerabilities in any of the installed software or hardware assets.
- For each vulnerability a description is provided along with affected assets.
- If available a solution is also provided with proof of the concept.

Process Input

- Input given to the application will vary based on the type of asset.
- Assets can be categorised into following categories:
  - Software
    - Licensed software
    - Free or open source software
    - In-house software
  - Hardware
    - Borrowed hardware
    - Owned hardware

Process Input

For example, inputs for software assets can be:

- Software type
- Software/Package vendor
- Software/Package name
- Date added
- Software Description
- Package ID
- Package version
- Severity score(if any vulnerability)
- Asset accessibility

Process Input

For example, inputs for hardware assets can be:

- Hardware type
- Hardware vendor
- Hardware model
- Date added
- Status
- Hardware ID
- Location of hardware
- Hardware Description

Processes Involved

Vulnerability Discovery

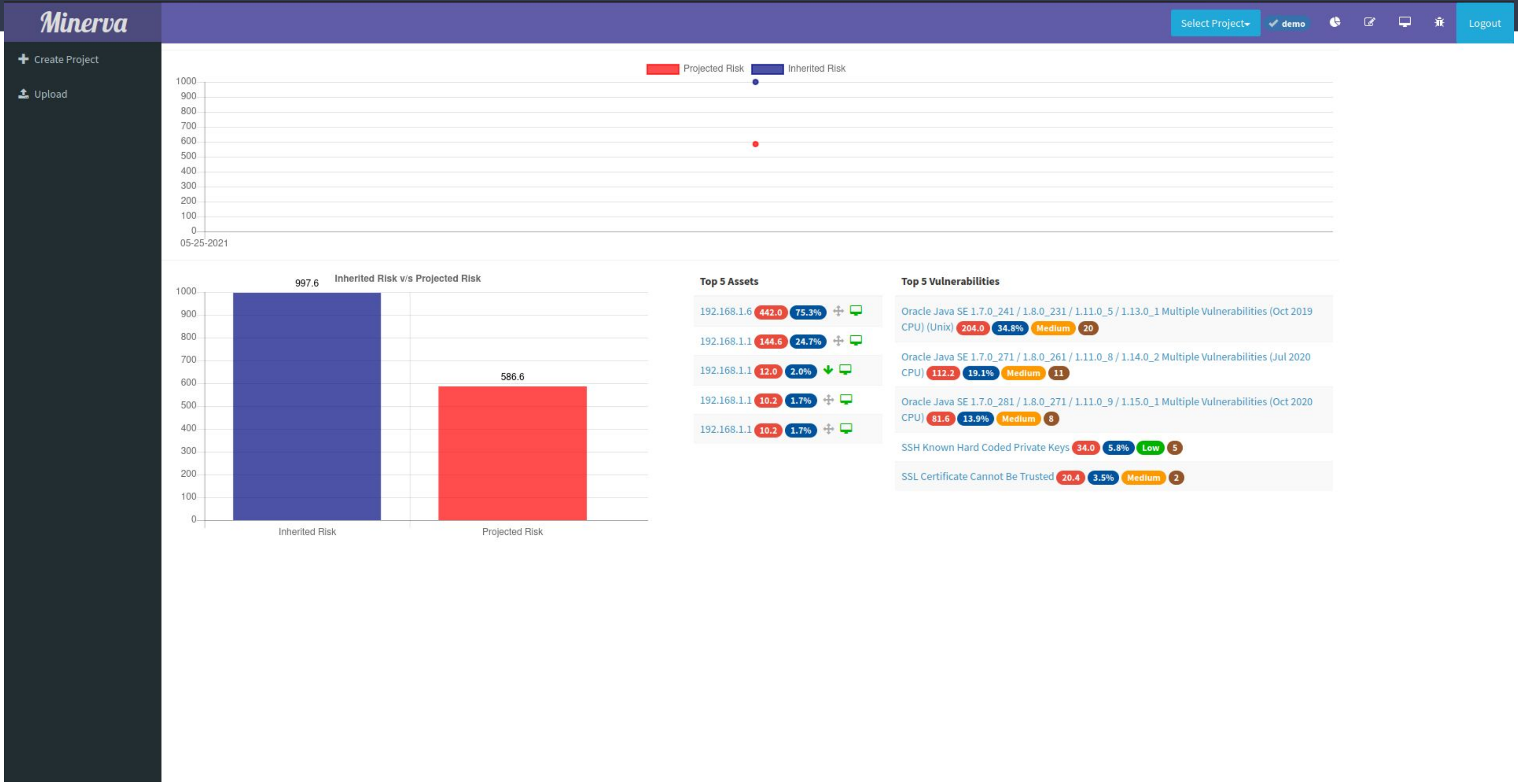
- Application searches the NVD(National Vulnerability Database) for any known unfixed vulnerabilities in any of the installed software or hardware assets.
- If a vulnerability is found, it takes it into account and map the vulnerability to the asset it has been found in.
- It also suggests a version of the software asset in which the said vulnerability is patched with proof of concept if available in the community driven NVD..

Processes Involved

Vulnerability Cause Detection:

- When an asset is added to the system, it gets added to a dependency chain.
- If asset A depends on asset B, then if asset B is vulnerable or fails or misconfigured then asset A will also be affected.
- As soon as a problem is discovered in the parent asset B, all the descendant assets A are marked affected with a reason referring to the asset B.

Results



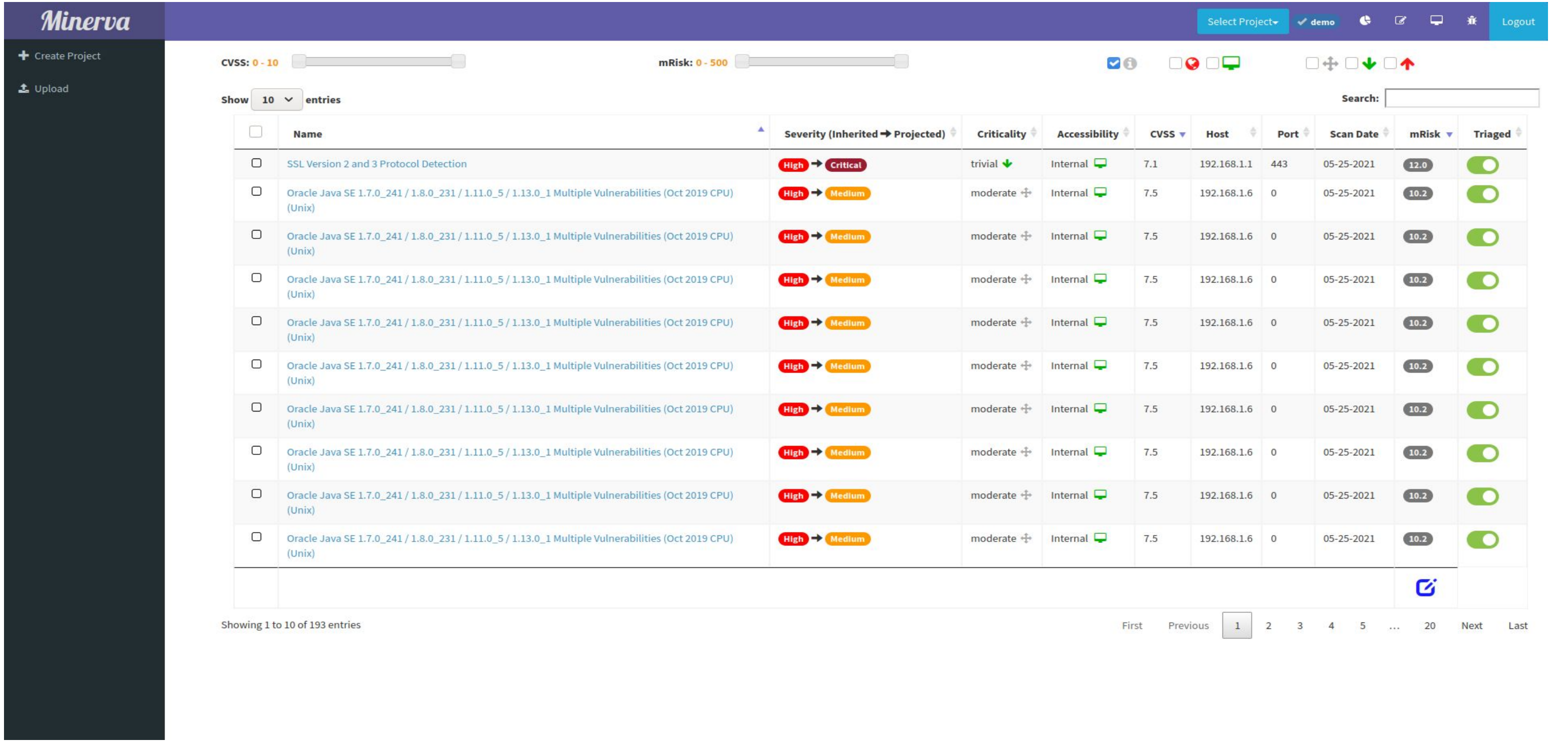
Top 5 Assets

- 192.168.1.6 442.0 75.3%
- 192.168.1.1 144.6 24.7%
- 192.168.1.1 12.0 2.0%
- 192.168.1.1 10.2 1.7%
- 192.168.1.1 10.2 1.7%

Top 5 Vulnerabilities

- Oracle Java SE 1.7.0_241 / 1.8.0_231 / 1.11.0_5 / 1.13.0_1 Multiple Vulnerabilities (Oct 2019 CPU) (Unix) 204.0 34.8% Medium 20
- Oracle Java SE 1.7.0_271 / 1.8.0_261 / 1.11.0_8 / 1.14.0_2 Multiple Vulnerabilities (Jul 2020 CPU) 112.2 19.1% Medium 11
- Oracle Java SE 1.7.0_281 / 1.8.0_271 / 1.11.0_9 / 1.15.0_1 Multiple Vulnerabilities (Oct 2020 CPU) 81.6 13.9% Medium 8
- SSH Known Hard Coded Private Keys 34.0 5.8% Low 5
- SSL Certificate Cannot Be Trusted 20.4 3.5% Medium 2

Results



CVSS: 0 - 10 mRisk: 0 - 500

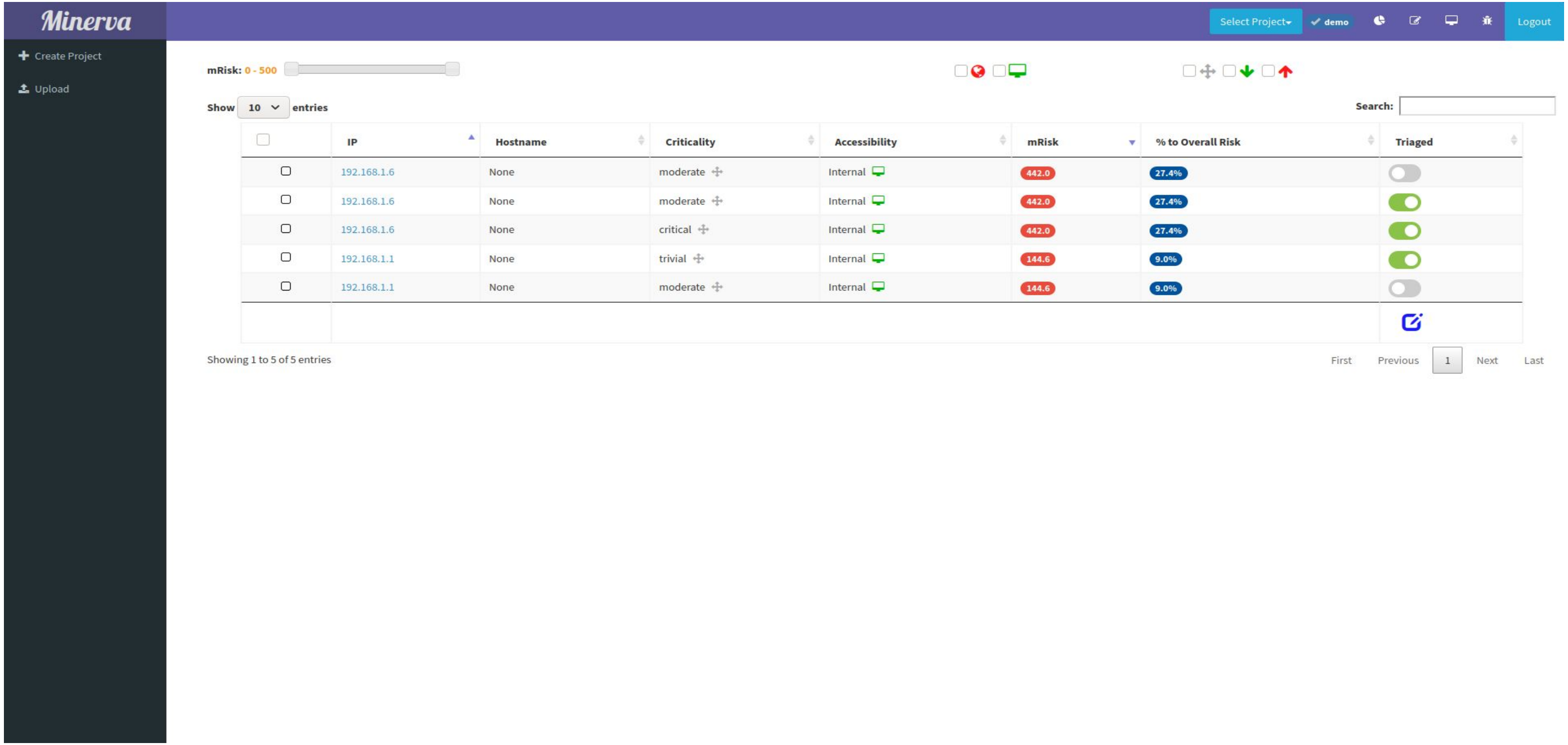
☑ ℹ □ □ 🖥 □ ✥ □ ↓ □ ↑

Show 10 entries Search:

□	Name	Severity (Inherited → Projected)	Criticality	Accessibility	CVSS	Host	Port	Scan Date	mRisk	Triaged
□	SSL Version 2 and 3 Protocol Detection	High → Critical	trivial ↓	Internal 🖥	7.1	192.168.1.1	443	05-25-2021	12.0	●
□	Oracle Java SE 1.7.0_241 / 1.8.0_231 / 1.11.0_5 / 1.13.0_1 Multiple Vulnerabilities (Oct 2019 CPU) (Unix)	High → Medium	moderate ✥	Internal 🖥	7.5	192.168.1.6	0	05-25-2021	10.2	●
□	Oracle Java SE 1.7.0_241 / 1.8.0_231 / 1.11.0_5 / 1.13.0_1 Multiple Vulnerabilities (Oct 2019 CPU) (Unix)	High → Medium	moderate ✥	Internal 🖥	7.5	192.168.1.6	0	05-25-2021	10.2	●
□	Oracle Java SE 1.7.0_241 / 1.8.0_231 / 1.11.0_5 / 1.13.0_1 Multiple Vulnerabilities (Oct 2019 CPU) (Unix)	High → Medium	moderate ✥	Internal 🖥	7.5	192.168.1.6	0	05-25-2021	10.2	●
□	Oracle Java SE 1.7.0_241 / 1.8.0_231 / 1.11.0_5 / 1.13.0_1 Multiple Vulnerabilities (Oct 2019 CPU) (Unix)	High → Medium	moderate ✥	Internal 🖥	7.5	192.168.1.6	0	05-25-2021	10.2	●
□	Oracle Java SE 1.7.0_241 / 1.8.0_231 / 1.11.0_5 / 1.13.0_1 Multiple Vulnerabilities (Oct 2019 CPU) (Unix)	High → Medium	moderate ✥	Internal 🖥	7.5	192.168.1.6	0	05-25-2021	10.2	●
□	Oracle Java SE 1.7.0_241 / 1.8.0_231 / 1.11.0_5 / 1.13.0_1 Multiple Vulnerabilities (Oct 2019 CPU) (Unix)	High → Medium	moderate ✥	Internal 🖥	7.5	192.168.1.6	0	05-25-2021	10.2	●
□	Oracle Java SE 1.7.0_241 / 1.8.0_231 / 1.11.0_5 / 1.13.0_1 Multiple Vulnerabilities (Oct 2019 CPU) (Unix)	High → Medium	moderate ✥	Internal 🖥	7.5	192.168.1.6	0	05-25-2021	10.2	●
□	Oracle Java SE 1.7.0_241 / 1.8.0_231 / 1.11.0_5 / 1.13.0_1 Multiple Vulnerabilities (Oct 2019 CPU) (Unix)	High → Medium	moderate ✥	Internal 🖥	7.5	192.168.1.6	0	05-25-2021	10.2	●
□	Oracle Java SE 1.7.0_241 / 1.8.0_231 / 1.11.0_5 / 1.13.0_1 Multiple Vulnerabilities (Oct 2019 CPU) (Unix)	High → Medium	moderate ✥	Internal 🖥	7.5	192.168.1.6	0	05-25-2021	10.2	●

Showing 1 to 10 of 193 entries

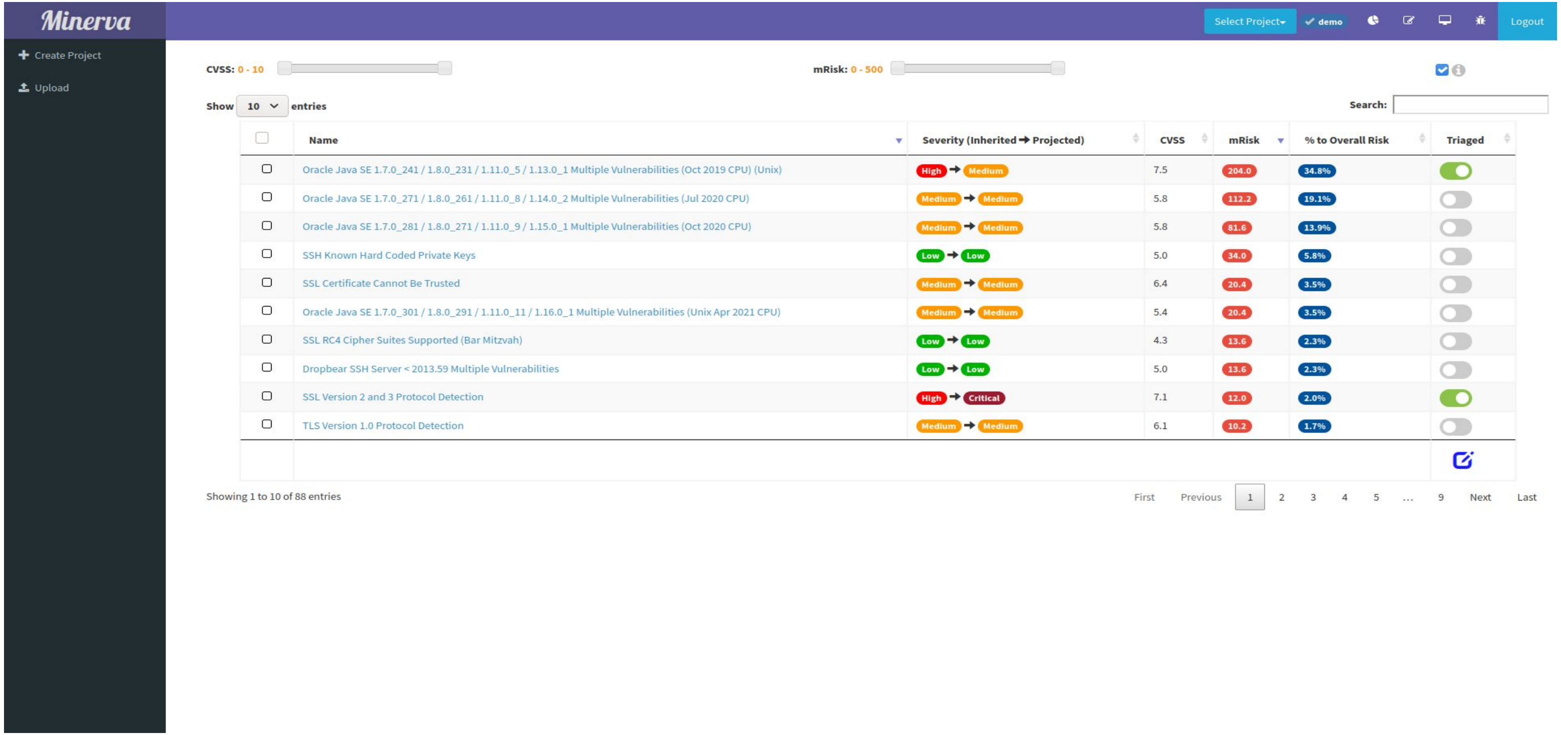
Results



□	IP	Hostname	Criticality	Accessibility	mRisk	% to Overall Risk	Triaged
□	192.168.1.6	None	moderate	Internal	442.0	27.4%	○
□	192.168.1.6	None	moderate	Internal	442.0	27.4%	●
□	192.168.1.6	None	critical	Internal	442.0	27.4%	●
□	192.168.1.1	None	trivial	Internal	144.6	9.0%	●
□	192.168.1.1	None	moderate	Internal	144.6	9.0%	○

Showing 1 to 5 of 5 entries

Results



CVSS: 0 - 10

mRisk: 0 - 500

Show 10 entries

Search:

□	Name	Severity (Inherited → Projected)	CVSS	mRisk	% to Overall Risk	Triaged
□	Oracle Java SE 1.7.0_241 / 1.8.0_231 / 1.11.0_5 / 1.13.0_1 Multiple Vulnerabilities (Oct 2019 CPU) (Unix)	High → Medium	7.5	204.0	34.8%	●
□	Oracle Java SE 1.7.0_271 / 1.8.0_261 / 1.11.0_8 / 1.14.0_2 Multiple Vulnerabilities (Jul 2020 CPU)	Medium → Medium	5.8	112.2	19.1%	○
□	Oracle Java SE 1.7.0_281 / 1.8.0_271 / 1.11.0_9 / 1.15.0_1 Multiple Vulnerabilities (Oct 2020 CPU)	Medium → Medium	5.8	81.6	13.9%	○
□	SSH Known Hard Coded Private Keys	Low → Low	5.0	34.0	5.8%	○
□	SSL Certificate Cannot Be Trusted	Medium → Medium	6.4	20.4	3.5%	○
□	Oracle Java SE 1.7.0_301 / 1.8.0_291 / 1.11.0_11 / 1.16.0_1 Multiple Vulnerabilities (Unix Apr 2021 CPU)	Medium → Medium	5.4	20.4	3.5%	○
□	SSL RC4 Cipher Suites Supported (Bar Mitzvah)	Low → Low	4.3	13.6	2.3%	○
□	Dropbear SSH Server < 2013.59 Multiple Vulnerabilities	Low → Low	5.0	13.6	2.3%	○
□	SSL Version 2 and 3 Protocol Detection	High → Critical	7.1	12.0	2.0%	●
□	TLS Version 1.0 Protocol Detection	Medium → Medium	6.1	10.2	1.7%	○

Showing 1 to 10 of 88 entries

First Previous 1 2 3 4 5 … 9 Next Last

Conclusion

The software developed enables us to understand the contextualized risk pertaining to each asset by each vulnerability across organization. It’s community based analytics provides a suggested risk for each vulnerability identified by vulnerability scanners and further strengthens risk prioritization process. So at any point of time teams can make an effective and more informed decision, based on unified and standardize data, about what vulnerabilities they should remediate (or can afford not to) on which asset(s).