

vPrioritizer

Art of Risk Prioritization

PRAMOD RANA - @IAmVarchashva

Manager, Application Security - Netskope

ABOUT ME

- Manager - Application Security @Netskope



- Security Testing & DevSecOps

- Author of three open source products:

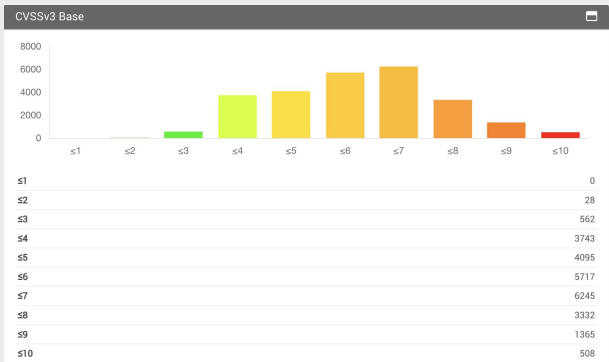
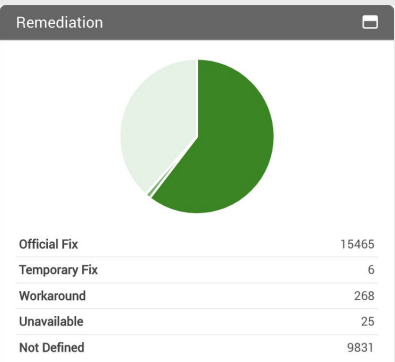
[Omniscient](#) - LetsMapYourNetwork: a graph-based asset management framework

[vPrioritizer](#) - Art of Risk Prioritization: a risk prioritization framework

[sec-depend-aid](#) - Dependabot Pull Request Monitoring Automation

- BlackHatEurope2018 | BlackHatUSA2019 | Defcon27 | BlackHatEurope2019 | nullconGoa2020 | BlackHatUSA2020 | OWASP Pune Chapter Leader | OSCP

CONTEXT



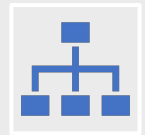
207'874
ENTRIES TOTAL

72
ADDED PER DAY Ø

124.3
UPDATED PER DAY Ø



As reflected, on a daily basis, ~70 new vulnerabilities become **known** to industry



Even if an organization considers the impact rate of 10%, it's still very challenging to manage it effectively



Huge number of vulnerabilities to *assess and remediate*; safe to assume that count is going to increase furthermore



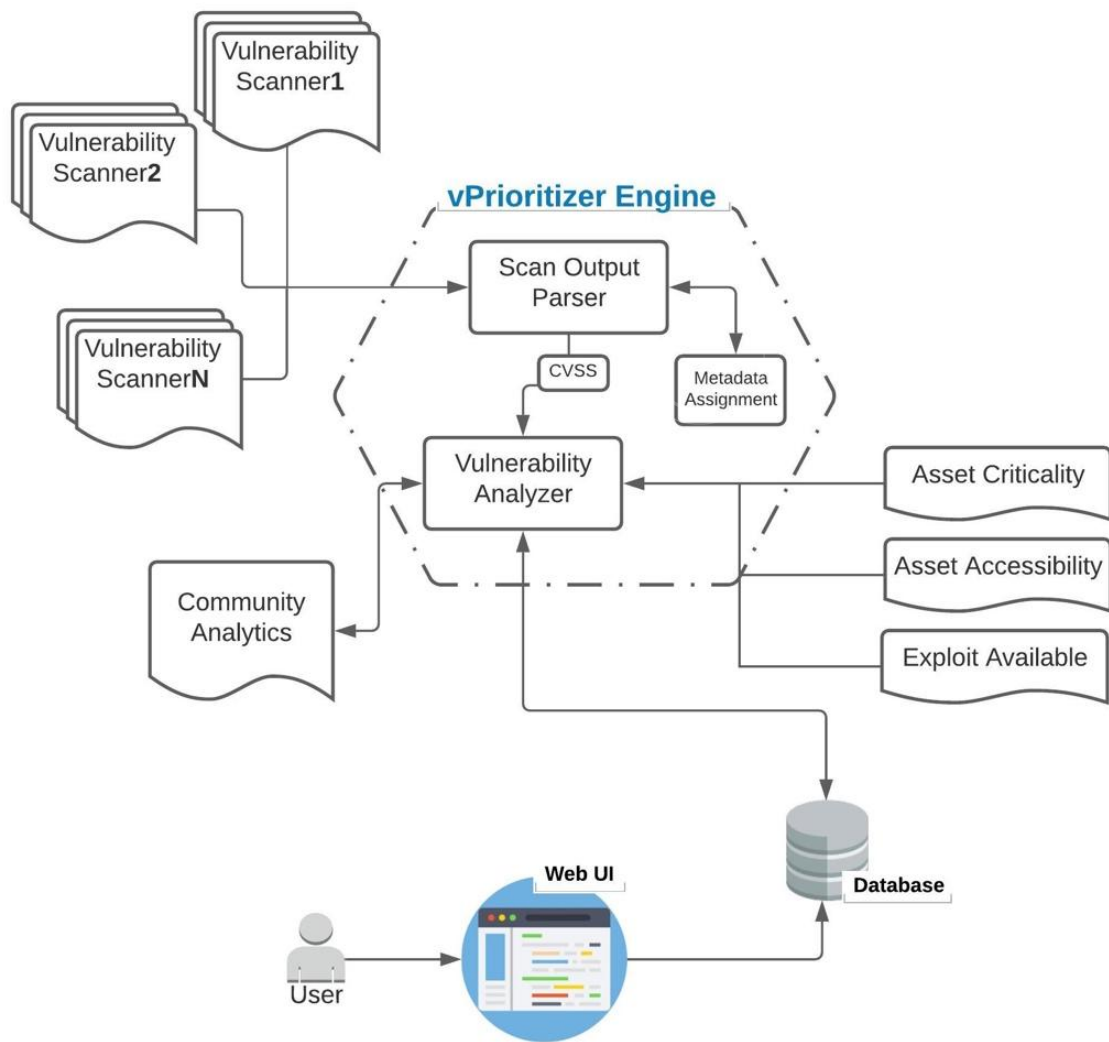
With this amount organization is focusing (or should focus) on **reducing** the risk rather than **eliminating** it

WHY vPRIORITIZER

In current era, vulnerability management is (almost) equal to risk prioritization because -

- Resources (skillset and time) is limited in every organization
- Environment is changing too fast and too frequently (ROI is less in analysis and remediation of a vulnerability if affected asset is not going to be live for a longer time - small attack surface)
- Attack surface is increasing exponentially in diversity (which again comes down to prioritization)
- Remember the 80/20 rule - **20% of vulnerabilities bring 80% of risk**

Risk is a contextualized value and depends on several factors like CVSS, exploit availability, asset criticality, asset availability etc. and practically difficult to determine across a medium to large organization.



HOW vPRIORITIZER WORKS

KEY FEATURES



Support upload of **csv** scan files from Nexpose, Nessus and QualysGuard (custom mapping option available **at runtime**)



User can assess the risk on different layers such as - significance on per asset basis, severity on per vulnerability basis, can adjust both factors at asset-vulnerability relationship level



Comprehensive dashboard containing multiple sections like "Inherited v/s Projected Risk (vPRisk)", "Top 5 vulnerabilities", "Top 5 Assets" & "Overall Program Timeline"



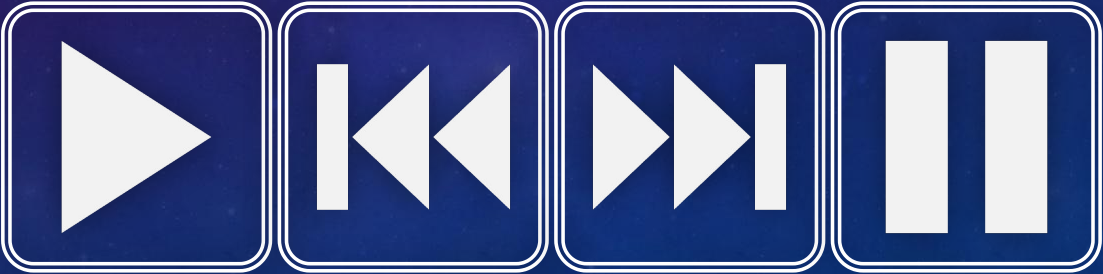
Project management feature



Docker support for Linux

CONCLUSION

- vPrioritizer enables us to understand the contextualized risk (vPRisk) pertaining to each asset by each vulnerability across the organization
- It's community-based analytics provides a suggested risk for each vulnerability identified by automated 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 standardized data, about what (vulnerability/ties) they should remediate (or can afford not to) and on which (asset/s).
- Risk is variable, function of multiple parameters
- Prioritize correctly - Fix first, what matters most
- Share with community - that's the only way to scale
- Fix fast, test fast, fail fast, adjust fast





<https://github.com/varchashva/vPrioritizer>



@IAmVarchashva



varchashva@gmail.com
rana.miet@gmail.com

