Automated Pen Testing as a Service (APTaaS™)

info@Horizon3.ai
www.Horizon3.ai
LinkedIn: Horizon3.ai
Twitter: @Horizon3ai

Vulnerable ≠ Exploitable

Criticality = ƒ(Exploitability, Impact)

Ever wonder how much of your time and effort is being wasted fixing things that don't actually matter?

You may be surprised to hear that a large majority of all vulnerabilities are unexploitable. According to data compiled by Kenna, in 2020, only 2.7% of the vulnerabilities found appeared to be exploitable and only 0.4% of those vulnerabilities were actually observed to be exploited at all.[1]

The traditional approaches of using agent-based vulnerability scanners and simplistic port-scans produce far too much noise, divert attention from the truly exploitable issues that represent provable risk to your business, and ignore how attackers really think and act. The prioritization of these low-risk or no-risk vulnerabilities alongside, or even above, the truly exploitable and impactful vulnerabilities can actually cause an organization's security posture to suffer. Meanwhile, more critical vulnerabilities are waiting in line for their turn to be remediated. If you can't properly prioritize, you will never secure your network.

So how do you know if it is critical to fix what you find?

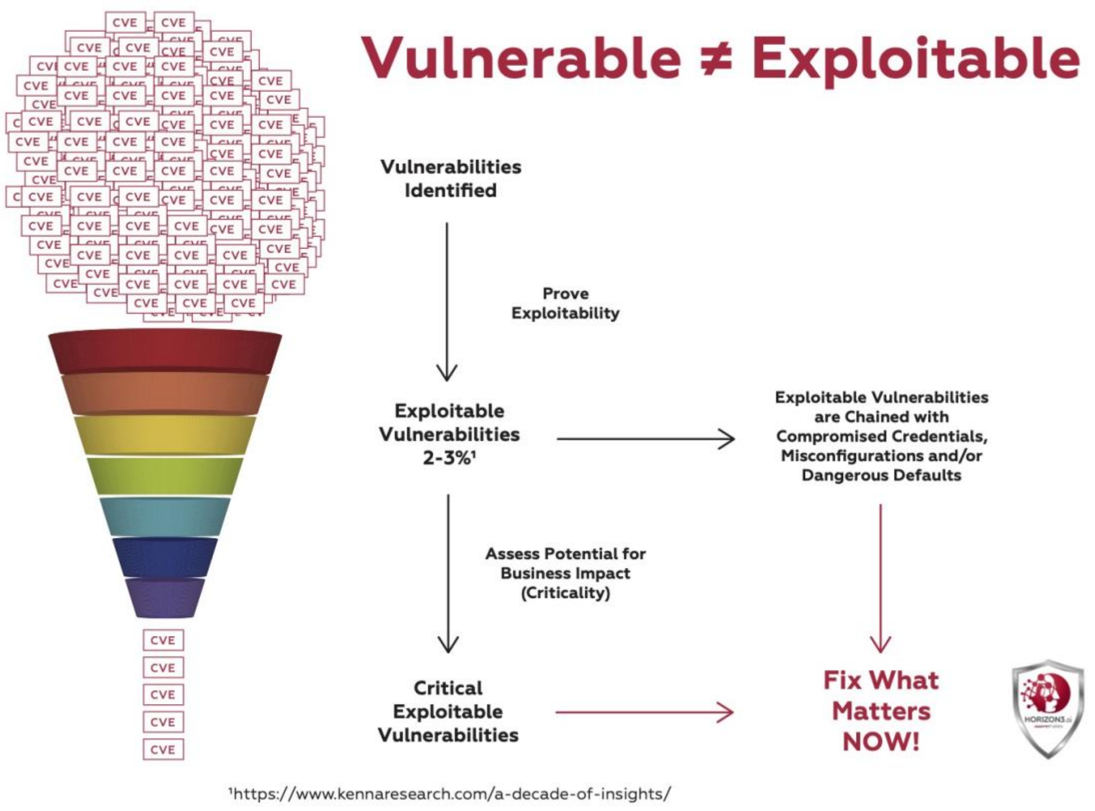
Criticality is a Function of Exploitability and Impact

The hardest part of cyber security is deciding what **not** to do because of limited time and resources. Spending valuable and scarce time and effort on remediating weaknesses that are not exploitable or do not represent a substantial business impact is itself a risk. At the very least, you should be able to trust that the findings from your security tools and services will appropriately guide your remediation and staffing decisions.

Criticality begins with the exploitability of a weakness. There are many reasons why a reported critical finding from vulnerability scanners and some pentesters may not be exploitable or would be very difficult to exploit, hence do not truly impose much or any risk.

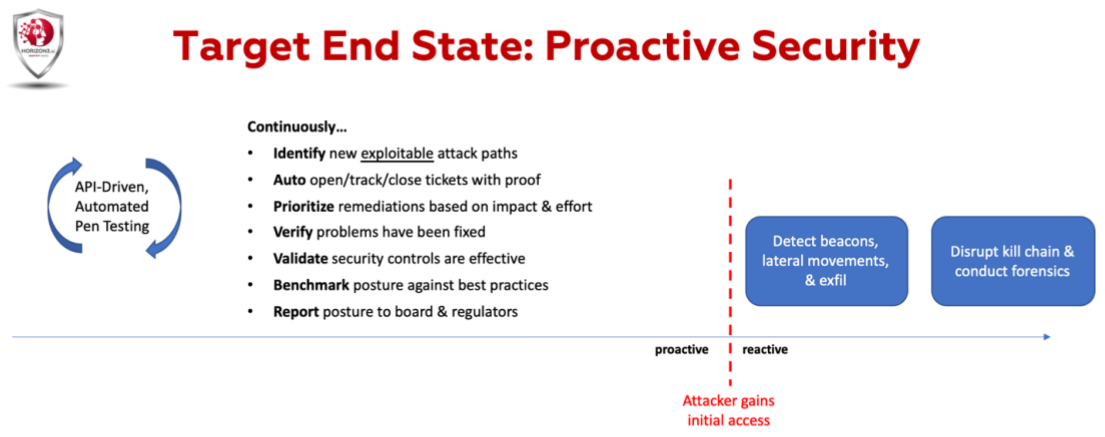
1. No exploit exists – There is no existing exploit available for the vulnerability.
2. High complexity – Several complex and/or impractical conditions must be met for the vulnerability to be exploited by an attacker.
3. Component not in use – The suspected software doesn't necessarily run in a vulnerable configuration.
4. Outdated ≠ exploitable – In the absence of a specific vulnerability, software being merely outdated/obsolete does not pose a critical risk.
5. Not accessible – The vulnerability exists in a part the software that isn't accessible from the attacker's perspective.
6. Network context – The context of where the vulnerable asset is in the network makes the risk informational rather than critical.

[1] https://www.kennaresearch.com/a-decade-of-insights/


Vulnerable ≠ Exploitable
CVE
Vulnerabilities Identified
Prove Exploitability
Exploitable Vulnerabilities 2-3%¹
Exploitable Vulnerabilities are Chained with Compromised Credentials, Misconfigurations and/or Dangerous Defaults
Assess Potential for Business Impact (Criticality)
Critical Exploitable Vulnerabilities
Fix What Matters NOW!
¹https://www.kennaresearch.com/a-decade-of-insights/

A Future of Continuous Security Assessment

Over the last decade, more and more CVEs/vulnerabilities are being found and reported, making it very hard to keep pace…it's snowballing and creating fatigue. With an annual manual pentest, you have giant craters in your security posture that develop between cycles as critical vulnerabilities come out; systems change with new software, patches and hardware; and personnel turns over.


Target End State: Proactive Security
API-Driven, Automated Pen Testing
Continuously…
Identify new exploitable attack paths
Auto open/track/close tickets with proof
Prioritize remediations based on impact & effort
Verify problems have been fixed
Validate security controls are effective
Benchmark posture against best practices
Report posture to board & regulators
Detect beacons, lateral movements, & exfil
Disrupt kill chain & conduct forensics
proactive
reactive
Attacker gains initial access

There is a need for a **proactive security posture** that includes continuous assessment, so you can catch up, keep up and even stay ahead.

Stay tuned for Part 2 – a deep dive into the Catch Up, Keep Up, Stay Ahead approach!

Vulnerable ≠ Exploitable
Criticality = f(Exploitability, Impact)