

Reading the Report: Triage, False Positives, and the Analyst Who Cried Wolf

Thu, Aug 13, 9:00 AM EDT · <https://scottslab.io/posts/vulnerability-scan-triage-and-correlation>

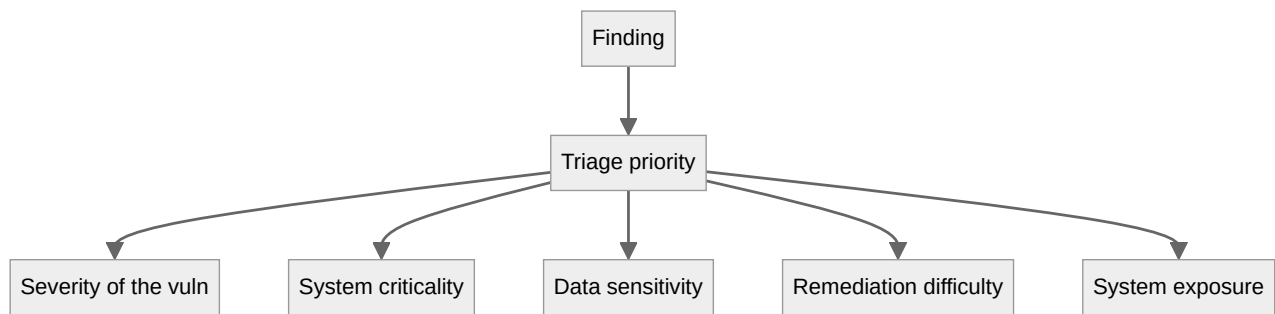


TL;DR

A scan report is a to-do list you have to rank, because you can't fix everything at once. Prioritize each finding on five factors: severity, system criticality, data sensitivity, remediation difficulty, and exposure. Validate before you escalate. One or two false positives and people stop trusting your tickets. Know the four outcomes (true/false positive, true/false negative; the false negative is the scary one), and correlate findings against compliance standards, your own config/log data, and historical trends.

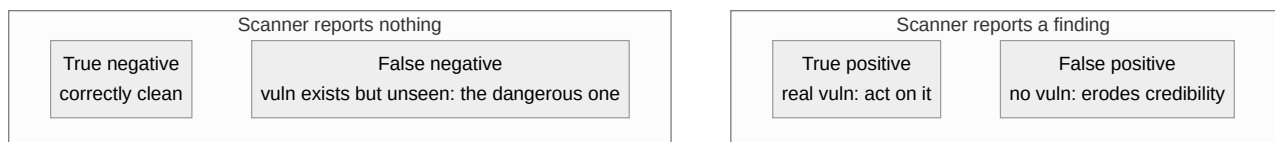
The scan finishes and hands you a pile of findings. The skill was never running the scan. It's what you do with the report, and the first job is ranking, because you can't remediate everything at once and pretending otherwise just parks the important fixes behind the trivial ones.

Five factors drive the triage. The severity of the vulnerability itself. The criticality of the affected system. The sensitivity of the data involved. The difficulty of remediation: a critical bug with a one-click patch may go ahead of a medium one that needs a maintenance window and a prayer. And the exposure of the system: an internet-facing host outranks the same flaw on an isolated internal box. Weigh those together and the pile sorts itself into an order that actually reduces risk instead of just closing tickets.



Reading the Report: Triage, False Positives, and the Analyst Who Cried Wolf

Before any of that reaches a system owner, validate. This is the step junior analysts skip and regret: escalate a false positive and you've burned someone else's time on nothing, do it twice and you're the analyst who cried wolf. After that your real findings get ignored, which is worse than not scanning at all. So confirm the finding is real by reviewing the scanner's own input/output evidence for it, and check whether there's already a compensating control or an accepted risk documented in the CMDB. A surprising share of "vulnerabilities" are things the org already knows about and has consciously chosen to live with.



Reading the Report: Triage, False Positives, and the Analyst Who Cried Wolf

It helps to hold the four outcomes clearly. A true positive is a real finding, correctly reported. A false positive is a reported finding that isn't actually there: annoying, credibility-eroding. A true negative is correctly finding nothing. And the dangerous one is the false negative: the scanner reports nothing but the vulnerability is really there. That's the risk you don't even know you're carrying, which is exactly why one scanner and one perspective are never enough.

Finally, findings mean more in context, so correlate them. Against compliance standards: PCI DSS, for one, says an external ASV scan can't pass with anything carrying a CVSS base score of 4.0 or higher (that's the numeric score, not CVSS v4.0 the version, and it's scoped to your internet-facing assets), which turns a "medium" on an exposed host into a hard fail. Against your own internal data: config-management systems and log repositories tell you whether a flagged service is even in use. And against history: the same XSS finding showing up scan after scan isn't one bug, it's a signal that a dev team needs training or an input-validation library, and you fix the pattern instead of the instance.