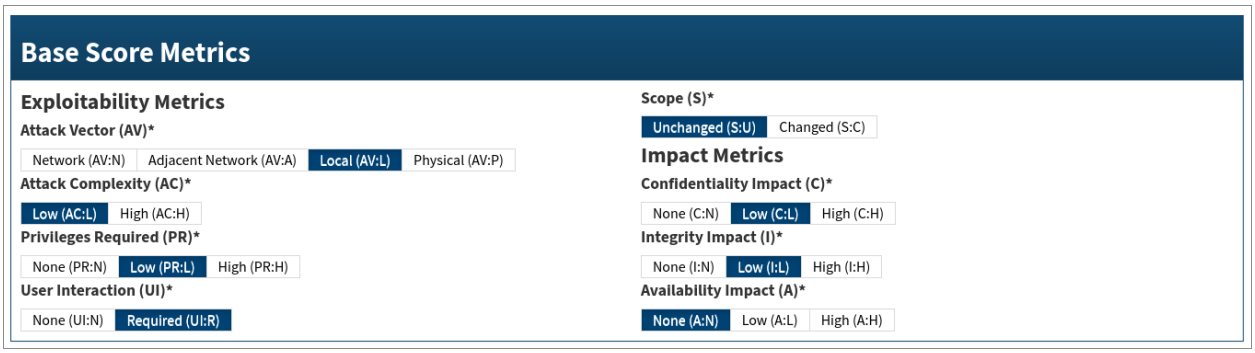
Ryan Schanzenbacher
12/1/23
CSEC-731

Risk Assessment



Overall, this vulnerability is going to be assigned a CVSS score of 3.9 (`AV:L/AC:L/PR:L/UI:R/S:U/C:L/I:L/A:N`) due to the following reasons:

1. Attack Vector - Local: Since this vulnerability is contingent on the attacker having some sort of access to the underlying operating system, you need to either have some sort of access, whether physical or over ssh, or some sort of insider, to execute this attack.
2. Attack Complexity - Low: Since the only requirement to execute this attack is a simple `ln -s` command, the complexity is considered low.
3. Privileges Required - Low: you need to be authenticated into the system to use this exploit, but you do not need to be privileged.
4. User Interaction - Required: This cannot be done without some sort of human intervention by itself, a human must run a script that sets up the environment or do it themselves.
5. Scope - Unchanged: By utilizing this exploit, the scope of permissions to the process or the user remain unchanged. Obviously this exploit is most dangerous when the server is run as root, either through Docker or manually.
6. Confidentiality Impact - Low: While any file that is symlinked to can be leaked to the public, it is contingent on a lot of acccess already, plus the scope of the file cannot be changed if the attacker loses access to the machine.
7. Integrity Impact - Low: As long as the user has write access to the symlinked file, certain methods can be used to change the data in the file. However, no other files out of scope can be modified. Also, the server option `--disable-mutations` can be used to mitigate this issue as well.
8. Availability Impact - None: No availability impact is observed because of this vulnerability

The biggest CWE this maps to is 22 - Improper Limitation of a Pathname to a Restricted Directory (https://cwe.mitre.org/data/definitions/22.html)

Of the ASVS tests, most of V12 from version 4 is violated either directly from this vulnerability or indirectly from other checks that are not performed. Per 12.1, no checks are performed on uploaded data, allowing a potential availability vulnerability if the disk is filled. Per 12.2, file type checks are not performed on uploaded data. Per 12.3 this is the best suited since most of these aren't applicable, only 12.3.1 is partially violated due to the issue above. Per 12.4, files are not scanned for viruses nor are they stored outside of the web root. My software is designed for everything to occur within the webroot, that is why the `--disable-mutations`

option exists. Per 12.5, uploaded files are treated the exact same as files that were present on startup. And, per 12.6, I do not implement ACLs.

Overall, while the vulnerability is *technically* a logic issue, there are so many pre-requisites to it being an issue that the viability of it actually being an issue is quite low, contributing to the 3.9 CVSS Score.