[root@reports]# docker exec -it clair-scanner sclair --allowlist allowlist.yaml singularity-images\_latest.sif

Found 45 Clair namespaces

Clair URL: http://0.0.0.0:6060/v1

1. Starting server...

======== Running on http://0.0.0.0:8080 ========

(Press CTRL+C to quit)

1. Checking server...

2. Processing images!

Exporting singularity-images\_latest.sif to targz...

singularity build --sandbox /tmp/tmptll9iapz/singularity-clair.3pgkj4s6 singularity-images\_latest.sif

INFO: Starting build...

INFO: Creating sandbox directory...

INFO: Build complete: /tmp/tmptll9iapz/singularity-clair.3pgkj4s6

INFO: Starting build...

INFO: Creating sandbox directory...

INFO: Build complete: /tmp/tmptll9iapz/singularity-clair.3pgkj4s6

...exported singularity-images\_latest.sif to /tmp/tmptll9iapz/singularity-clair.3pgkj4s6.gz

...serving http://0.0.0.0:8080/images/singularity-clair.3pgkj4s6.gz to Clair

3. Generating report!

Allowlist: skipping CVE-2019-9924

Allowlist: skipping CVE-2019-13627

Allowlist: skipping CVE-2019-5188

Allowlist: skipping CVE-2019-5094

Allowlist: skipping CVE-2017-18269

Allowlist: skipping CVE-2018-11237

Allowlist: skipping CVE-2017-12133

Allowlist: skipping CVE-2019-9169

Allowlist: skipping CVE-2018-6485

Allowlist: skipping CVE-2009-5155

Allowlist: skipping CVE-2018-11236

Allowlist: skipping CVE-2019-19126

Allowlist: skipping CVE-2016-9843

Allowlist: skipping CVE-2016-9842

Allowlist: skipping CVE-2016-9841

Allowlist: skipping CVE-2016-9840

Allowlist: skipping CVE-2018-20482

Allowlist: skipping CVE-2019-9923

Allowlist: skipping CVE-2020-1712

Allowlist: skipping CVE-2019-20386

Allowlist: skipping CVE-2018-16888

Allowlist: skipping CVE-2016-3189

Allowlist: skipping CVE-2019-12900

bash - 4.3-14ubuntu1.3

----------------------

libgcrypt20 - 1.6.5-2ubuntu0.5

------------------------------

perl - 5.22.1-9ubuntu0.6

------------------------

CVE-2020-10543 (Low) unapproved

http://people.ubuntu.com/~ubuntu-security/cve/CVE-2020-10543

Perl before 5.30.3 on 32-bit platforms allows a heap-based buffer overflow because nested regular expression quantifiers have an integer overflow. An application written in Perl would only be vulnerable to this flaw if it evaluates regular expressions supplied by the attacker. Evaluating regular expressions in this fashion is known to be dangerous since the regular expression engine does not protect against denial of service attacks in this usage scenario. Additionally, the target system needs a sufficient amount of memory to allocate partial expansions of the nested quantifiers prior to the overflow occurring. This requirement is unlikely to be met on 64bit systems.]

CVE-2020-10878 (Low) unapproved

http://people.ubuntu.com/~ubuntu-security/cve/CVE-2020-10878

Perl before 5.30.3 has an integer overflow related to mishandling of a "PL\_regkind[OP(n)] == NOTHING" situation. A crafted regular expression could lead to malformed bytecode with a possibility of instruction injection. An application written in Perl would only be vulnerable to this flaw if it evaluates regular expressions supplied by the attacker. Evaluating regular expressions in this fashion is known to be dangerous since the regular expression engine does not protect against denial of service attacks in this usage scenario.]

CVE-2020-12723 (Low) unapproved

http://people.ubuntu.com/~ubuntu-security/cve/CVE-2020-12723

regcomp.c in Perl before 5.30.3 allows a buffer overflow via a crafted regular expression because of recursive S\_study\_chunk calls. An application written in Perl would only be vulnerable to this flaw if it evaluates regular expressions supplied by the attacker. Evaluating regular expressions in this fashion is known to be dangerous since the regular expression engine does not protect against denial of service attacks in this usage scenario.]

e2fsprogs - 1.42.13-1ubuntu1

----------------------------

glibc - 2.23-0ubuntu11

----------------------

CVE-2020-10029 (Low) unapproved

http://people.ubuntu.com/~ubuntu-security/cve/CVE-2020-10029

The GNU C Library (aka glibc or libc6) before 2.32 could overflow an on-stack buffer during range reduction if an input to an 80-bit long double function contains a non-canonical bit pattern, a seen when passing a 0x5d414141414141410000 value to sinl on x86 targets. This is related to sysdeps/ieee754/ldbl-96/e\_rem\_pio2l.c.

CVE-2020-6096 (Low) unapproved

http://people.ubuntu.com/~ubuntu-security/cve/CVE-2020-6096

An exploitable signed comparison vulnerability exists in the ARMv7 memcpy() implementation of GNU glibc 2.30.9000. Calling memcpy() (on ARMv7 targets that utilize the GNU glibc implementation) with a negative value for the 'num' parameter results in a signed comparison vulnerability. If an attacker underflows the 'num' parameter to memcpy(), this vulnerability could lead to undefined behavior such as writing to out-of-bounds memory and potentially remote code execution. Furthermore, this memcpy() implementation allows for program execution to continue in scenarios where a segmentation fault or crash should have occurred. The dangers occur in that subsequent execution and iterations of this code will be executed with this corrupted data.

CVE-2020-1752 (Low) unapproved

http://people.ubuntu.com/~ubuntu-security/cve/CVE-2020-1752

A use-after-free vulnerability introduced in glibc upstream version 2.14 was found in the way the tilde expansion was carried out. Directory paths containing an initial tilde followed by a valid username were affected by this issue. A local attacker could exploit this flaw by creating a specially crafted path that, when processed by the glob function, would potentially lead to arbitrary code execution. This was fixed in version 2.32.

CVE-2020-1751 (Medium) unapproved

http://people.ubuntu.com/~ubuntu-security/cve/CVE-2020-1751

An out-of-bounds write vulnerability was found in glibc before 2.31 when handling signal trampolines on PowerPC. Specifically, the backtrace function did not properly check the array bounds when storing the frame address, resulting in a denial of service or potential code execution. The highest threat from this vulnerability is to system availability.

apt - 1.2.32

------------

CVE-2020-3810 (Medium) unapproved

http://people.ubuntu.com/~ubuntu-security/cve/CVE-2020-3810

Missing input validation in the ar/tar implementations of APT before version 2.1.2 could result in denial of service when processing specially crafted deb files.

CVE-2020-27350 (Medium) unapproved

http://people.ubuntu.com/~ubuntu-security/cve/CVE-2020-27350

APT had several integer overflows and underflows while parsing .deb packages, aka GHSL-2020-168 GHSL-2020-169, in files apt-pkg/contrib/extracttar.cc, apt-pkg/deb/debfile.cc, and apt-pkg/contrib/arfile.cc. This issue affects: apt 1.2.32ubuntu0 versions prior to 1.2.32ubuntu0.2; 1.6.12ubuntu0 versions prior to 1.6.12ubuntu0.2; 2.0.2ubuntu0 versions prior to 2.0.2ubuntu0.2; 2.1.10ubuntu0 versions prior to 2.1.10ubuntu0.1;

zlib - 1:1.2.8.dfsg-2ubuntu4.1

------------------------------

tar - 1.28-2.1ubuntu0.1

-----------------------

systemd - 229-4ubuntu21.21

--------------------------

bzip2 - 1.0.6-8

here is the list of unapproved vulnerabilities that sneaked in inspite of adding all vulnerabilities available in JSON file

[root@reports]# docker exec -it clair-scanner sclair --allowlist allowlist.yaml singularity-images\_latest.sif | grep -i unapproved

CVE-2020-3810 (Medium) unapproved

CVE-2020-27350 (Medium) unapproved

CVE-2020-10029 (Low) unapproved

CVE-2020-6096 (Low) unapproved

CVE-2020-1752 (Low) unapproved

CVE-2020-1751 (Medium) unapproved

CVE-2020-10543 (Low) unapproved

CVE-2020-10878 (Low) unapproved

CVE-2020-12723 (Low) unapproved