

**Step-1** > First, ensure you have the curl installed.

```
sudo apt-get update
```

```
(appsec@kali-appsec)-[~]
$ sudo apt-get update
Ign:2 https://download.docker.com/linux/debian kali-rolling InRelease
Get:1 http://kali.download/kali kali-rolling InRelease [41.5 kB]
Err:3 https://download.docker.com/linux/debian kali-rolling Release
404 Not Found [IP: 2600:9000:2041:ca00:3:db06:4200:93a1 443]
Get:4 http://kali.download/kali kali-rolling/main amd64 Packages [20.6 MB]
Get:5 http://kali.download/kali kali-rolling/main amd64 Contents (deb) [49.2 MB]
Get:6 http://kali.download/kali kali-rolling/contrib amd64 Packages [115 kB]
Get:7 http://kali.download/kali kali-rolling/contrib amd64 Contents (deb) [267 kB]
Get:8 http://kali.download/kali kali-rolling/non-free amd64 Packages [202 kB]
Get:9 http://kali.download/kali kali-rolling/non-free amd64 Contents (deb) [884 kB]
Get:10 http://kali.download/kali kali-rolling/non-free-firmware amd64 Packages [10.8 kB]
Get:11 http://kali.download/kali kali-rolling/non-free-firmware amd64 Contents (deb) [24.3 kB]
Reading package lists... Done
E: The repository 'https://download.docker.com/linux/debian kali-rolling Release' does not have a Release file.
N: Updating from such a repository can't be done securely, and is therefore disabled by default.
N: See apt-secure(8) manpage for repository creation and user configuration details.
```

```
sudo apt-get install -y curl
```

```
(appsec@kali-appsec)-[~]
$ sudo apt-get install -y curl
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libssl-dev libtirpc-dev
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  cryptsetup cryptsetup-bin cryptsetup-initramfs cryptsetup-nuke-password gnutls-bin ldap-utils libc-bin libc-dev-bin libc-devtools libc-l10n libc6 libc6-dev libc6-i386 libcryptsetup12 libcurl3t64-gnutls libgnutls-dane0t64
  libgnutls14t64 libhogweed6t64 libldap-common libldap2 libnettle0t64 libnghttp2-9 libnghttp2-16 libnghttp2-crypto-gnutls8 libnss-systemd libpam-systemd libpsl5t64 libssh2-1t64 libssl1t64 libsystemd-shared libsystemd0 libudev1
  linux-base linux-sysctl-defaults locales openssl openssl-provider-legacy systemd systemd-cryptsetup systemd-dev systemd-sysv systemd-timesyncd udev
Suggested packages:
  libssl2-modules-gsapi-mit | libssl2-modules-gsapi-heimdal glibc-doc libnss-nis libnss-nisplus libtss2-rc0t64 libarchive13t64 libdw1t64 libelf1t64 system-container systemd-homed systemd-userdbd systemd-boot systemd-resolved
  systemd-repart
The following packages will be REMOVED:
  libcurl3-gnutls libgnutls-dane0 libgnutls13 libhogweed6 libnettle0 libpsl5 libssh2-1 libssl3
The following NEW packages will be installed:
  libcurl3t64-gnutls libgnutls-dane0t64 libgnutls13t64 libhogweed6t64 libldap2 libnettle0t64 libnghttp2-9 libnghttp2-16 libnghttp2-crypto-gnutls8 libpsl5t64 libssh2-1t64 libssl1t64 libsystemd-shared libsystemd0 libudev1
  linux-base locales openssl systemd systemd-dev systemd-sysv systemd-timesyncd udev
The following packages will be upgraded:
  cryptsetup cryptsetup-bin cryptsetup-initramfs cryptsetup-nuke-password curl gnutls-bin ldap-utils libc-bin libc-dev-bin libc-devtools libc-l10n libc6 libc6-dev libc6-i386 libcryptsetup12 libldap-common libnss-systemd
  libpam-systemd libsystemd-shared libsystemd0 libudev1 linux-base locales openssl systemd systemd-dev systemd-sysv systemd-timesyncd udev
```

**Step-2** > Install Docker compose & docker.io and check their version.

```
sudo apt-get install docker-compose
```

```
$ sudo apt-get install docker-compose
[sudo] password for kausik:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  fonts-inter-variable gnome-accessibility-themes gnome-themes-extra
  gnome-themes-extra-data icu-devtools libflac12t64 libgail-common
  libgail18t64 libgeos3.13.0 libglapi-mesa libgtk2.0-0t64 libgtk2.0-bin
  libgtk2.0-common libicu-dev liblbfgsb0 libpoppler145 libpython3.12-minimal
  libpython3.12-stdlib libpython3.12t64 python3-setproctitle python3.12-tk
  ruby-zeitwerk
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  containerd criu docker-buildx docker-cli docker.io libcompel1 libintl-perl
  libintl-xs-perl libmodule-find-perl libproc-processtable-perl
  libsort-naturally-perl needrestart python3-compose python3-docker
  python3-dockerpty python3-pycrui python3-texttable runc tini
Suggested packages:
  containernetworking-plugins docker-doc aufs-tools btrfs-progs cgroupfs-mount
  debootstrap rinse rootlesskit xfsprogs zfs-fuse | zfsutils-linux
The following NEW packages will be installed:
  containerd criu docker-buildx docker-cli docker-compose docker.io libcompel1
  libintl-perl libintl-xs-perl libmodule-find-perl libproc-processtable-perl
  libsort-naturally-perl needrestart python3-compose python3-docker
  python3-dockerpty python3-pycrui python3-texttable runc tini
0 upgraded, 20 newly installed, 0 to remove and 10 not upgraded.
Need to get 81.5 MB of archives.
After this operation, 335 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
```

```
sudo apt-get install docker.io
```

```
└─$ sudo apt-get install docker.io
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
docker.io is already the newest version (26.1.5+dfsg1-9+b2).
docker.io set to manually installed.
The following packages were automatically installed and are no longer required:
 fonts-inter-variable gnome-accessibility-themes gnome-themes-extra
 gnome-themes-extra-data icu-devtools libflac12t64 libgail-common
 libgail18t64 libgeos3.13.0 libglapi-mesa libgtk2.0-0t64 libgtk2.0-bin
 libgtk2.0-common libicu-dev liblbfgsb0 libpoppler145 libpython3.12-minimal
 libpython3.12-stdlib libpython3.12t64 python3-setproctitle python3.12-tk
 ruby-zeitwerk
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 10 not upgraded.
```

```
docker --version
```

```
docker.io --version
```

```
└─(kausik@kali)-[~]
└─$ docker --version
Docker version 26.1.5+dfsg1, build a72d7cd

└─(kausik@kali)-[~]
└─$ docker-compose --version
docker-compose version 1.29.2, build unknown
```

**Step-3** > To configure and deploy OWASP crAPI, follow these steps:

1. Download the docker-compose.yml file:

```
sudo curl -o docker-compose.yml https://raw.githubusercontent.com/OWASP/crAPI/main/deploy/docker/docker-compose.yml
```

```
(kausik@kali)-[~]
└─$ curl -o docker-compose.yml https://raw.githubusercontent.com/OWASP/crAPI/main/deploy/docker/docker-compose.yml
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100  7512  100  7512    0     0   7377      0  0:00:01  0:00:01 --:--:--  7379
```

2. Pull the necessary Docker images:

```
sudo docker-compose pull
```

```
(kausik@kali)-[~]
└─$ sudo docker-compose pull
Pulling postgresdb ... done
Pulling mongodb ... done
Pulling mailhog ... done
Pulling crapi-identity ... done
Pulling crapi-community ... done
Pulling crapi-workshop ... done
Pulling crapi-web ... done
Pulling api.mypremiumdealership.com ... done
```

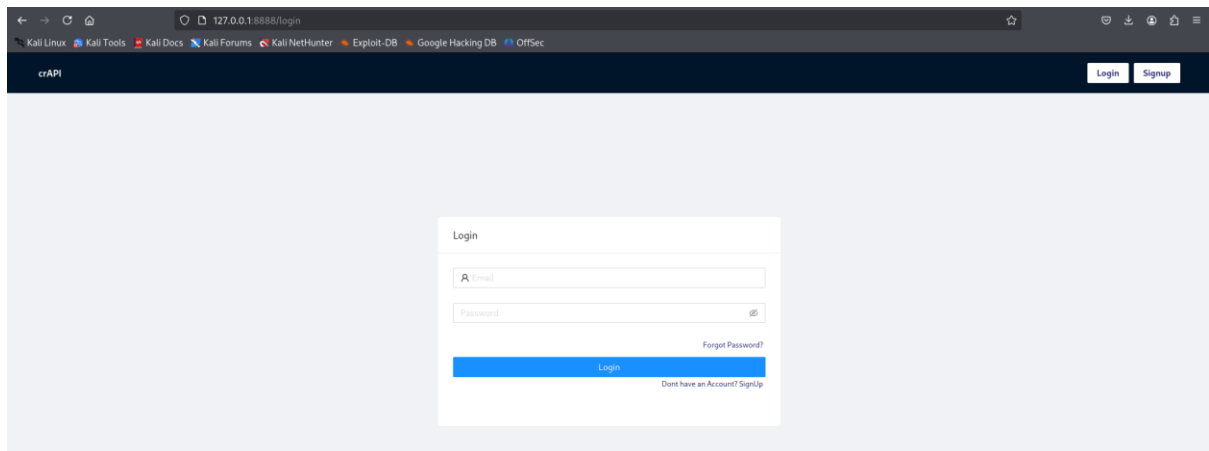
3. Start the OWASP crAPI application using docker-compose: [Every time you have to run this command for getting crAPI open in browser after shutdown/restart of the system.]

```
sudo docker-compose -f docker-compose.yml --compatibility up -d
```

```
(kausik@kali)-[~]
└─$ sudo docker-compose -f docker-compose.yml --compatibility up -d
Creating network "kausik_default" with the default driver
Creating volume "kausik_mongodb-data" with default driver
Creating volume "kausik_postgresql-data" with default driver
Creating api.mypremiumdealership.com ... done
Creating mongodb ... done
Creating mailhog ... done
Creating postgresdb ... done
Creating crapi-identity ... done
Creating crapi-community ... done
Creating crapi-workshop ... done
Creating crapi-web ... done
```

**Step-4** > To open OWASP crAPI in local browser:

Visit <http://localhost:8888>



**Step-5** > To open OWASP crAPI MailHog in local browser:

Visit <http://localhost:8025>

