



Troubleshooting

- [Docker Pull Issues, on page 1](#)

Docker Pull Issues

Default Secure Configuration

By default, the system pulls the Docker image using the Center's FQDN with TLS certificate verification. Configure two settings in this scenario:

1. **Name Resolution:** Resolve the FQDN through a reachable DNS server or a local configuration.
2. **Certificate:** Ensure the Center and Docker access the same certificate authority, or manually share the Center's certificate.

Observe name resolution issues during Docker container creation.

```
escalation@escalation-SENSORS:~/Documents$ sudo docker compose up
[+] Running 1/0
 x ccv-sensor-1 Error Get "https://Center162:443/v2/": dial tcp: lookup Center162 on 127.0.0.1:53: server misbehaving
Error response from daemon: Get "https://Center162:443/v2/": dial tcp: lookup Center162 on 127.0.0.1:53: server misbehaving
escalation@escalation-SENSORS:~/Documents$
```

If there is a certificate issue, the following problem can occur:

```
escalation@sensor-esc-01:~/Documents$ sudo docker compose up
[+] Running 3/0
 x ccv-sensor-active Error context canceled 0.0s
 x ccv-sensor-1 Error context canceled 0.0s
 x ccv-sensor-2 Error Get "https://CenterDoc1-65:443/v2/": tls: failed to verify certificate: x509: certificate signed by unknown authority 0.0s
Error response from daemon: Get "https://CenterDoc1-65:443/v2/": tls: failed to verify certificate: x509: certificate signed by unknown authority
escalation@sensor-esc-01:~/Documents$
```

This means that the certificate found is incorrect, or the folder containing the local certificate does not have the correct name.

Default Insecure Configuration

If the Center's FQDN and TLS certificate verification are not usable, pull the Docker image using the Center's IP address directly. Add an exception on the Docker host to allow insecure registry usage.

If the option to allow insecure registry usage is not properly configured, the following message appears:

```
escalation@sensor-esc-01:~/Documents$ sudo docker compose up
[+] Running 3/3
x ccv-sensor-2 Error      Get "https://192.168.49....           0.1s
x ccv-sensor-active Error context canceled           0.1s
x ccv-sensor-1 Error      context canceled           0.1s
Error response from daemon: Get "https://192.168.49.33:443/v2/": tls: failed to verify certificate: x509: cannot validate certificate for 192.168.49.33 because it doesn't contain any IP SANs
escalation@sensor-esc-01:~/Documents$
```

Error response from daemon: Get "https://192.168.49.30:443/v2/": tls: failed to verify certificate: x509: cannot validate certificate for 192.168.49.30 because it doesn't contain any IP SANs.

In this case, configure the system to allow insecure registry usage. Add the Center's IP to `/etc/docker/daemon.json` and restart Docker with `sudo systemctl restart docker.service`.

For example:

```
escalation@escalation-docker02:~/Documents$ cat /etc/docker/daemon.json
{
    "insecure-registries" : [ "192.168.49.30:443" ]
}
escalation@escalation-docker02:~/Documents$
```

Deployment Token Issue

A deployment token issue prevents the sensor from starting. Observe the sensor logs in the Docker container during the sensor startup. For example:

```
escalation@sensor-esc-01:~/Documents$ sudo docker compose up
[+] Running 1/1
✓ ccv-sensor-1 Pulled
0.2s
WARN[0000] Found orphan containers ([ccv-sensor-2 ccv-sensor-active]) for this project. If you removed or renamed this service in your compose file, you can run this command with the --remove-orphans flag to clean it up.
[+] Running 1/0
✓ Container ccv-sensor-1 Created
0.0s
Attaching to ccv-sensor-1
ccv-sensor-1 | 10/12/2024 10:52:15 gosh INFO creating symlink /data/etc/ca-certificates.crt->/etc/ssl/certs/ca-certificates.crt [caller=cert.go:48]
ccv-sensor-1 | 10/12/2024 10:52:15 gosh INFO Mount point size is 61582741504, threshold is 1063256064 [caller=mount.go:30]
ccv-sensor-1 | 10/12/2024 10:52:15 gosh INFO persistent data '/data' dir looks big enough [caller=bootstrap_nux.go:32]
ccv-sensor-1 | 10/12/2024 10:52:16 gosh ERROR Auto enroll failed: error during enroll: got status 403 [caller=environment.go:378]
ccv-sensor-1 | 10/12/2024 10:52:16 gosh INFO no provisioning package found in /data/appdata/sbs-sensor-config-Docker01-enp6s0.zip: sleeping [caller=environment.go:383]
```

- ccv-sensor-1 | 10/12/2024 10:52:16 gosh ERROR Auto enroll failed: error during enroll: got status 403 [caller=environment.go:378]
- ccv-sensor-1 | 10/12/2024 10:52:16 gosh INFO no provisioning package found in /data/appdata/sbs-sensor-config-Docker01-enp6s0.zip: sleeping [caller=environment.go:383]

In the Cisco Cyber Vision Center user interface, some messages can help with troubleshooting. The Sensor Explorer page displays a following message if the deployment token is no longer valid or has already been used.

Warning: A deployment using a deployment token has failed. Please verify the token's validity in the deployment token section.

Acknowledge

The deployment token page identifies other causes. For example, a token that is already used displays the message: "invalid nonce."

Name	Tokens	Status	Creation Date	Expiration Date	Usages	Last deployment	Actions
TokensDepl-01	Image	Enabled	Oct 30, 2024	Oct 14, 2026	11/102	<div>Deployment unsuccessful: invalid nonce</div> <div>Last failed attempt: Dec 3, 2024</div>	<div> <div>Show</div> <div></div> </div>
	cviox-aarch64.tar						
	cviox-active-discovery-aarch64.tar						
	cviox-active-discovery-ic3000-x86-64.tar						
	cviox-active-discovery-x86-64.tar						
	cviox-ic3000-x86-64.tar						
	cviox-x86-64.tar						

Docker Issues with Ethernet Boards

Some Ethernet boards are not supported, as explained in the prerequisites. The Cisco Cyber Vision Docker sensor cannot use them as DPI interfaces because they do not support the applied Docker network configuration.

Cisco tests different vendors and recommends using Intel Ethernet adapters. It does not support USB adapters.

An incompatible Ethernet board produces error messages during Docker container creation, such as:

- Error response from daemon: failed to create task for container: failed to create shim task: OCI runtime create failed: runc create failed: unable to start container process: error during container init: error running hook #0: error running hook: exit status 1, stdout: , stderr: failed to add interface vethd56cba0 to sandbox: error setting interface "vethd56cba0" MAC to "02:42:ac:13:00:02": device or resource busy: unknown

