Mandiant

TROUBLESHOOT PROTECTED THEATER

If you are having issues registering your Protected Theater or Protected Actor, there are several tools available to help identify and resolve the issue. These include:

- **Monitoring Registration Communication**
- **Addressing Communication Issues during Registration**
- **Restart Services**

Monitoring Registration Communication

You can monitor the registration process by verifying network information (interface IP addresses, route tables, etc.) start getting updated in the Director. The file you monitor depends on where you are monitoring.

When available, log samples are provided. IP addresses may have been scrubbed, using **xxx** to replace private values. Long lines may have been truncated, with **...** added to the end of a line to show that there was additional data.

- Director: `/opt/apps/verodin/planner/log/verodin_node_log`

```
2018-05-14 13:25:36 +0000 : NODES FOR CONNECT CHECKING: []
-------
2018-05-14 13:25:36 +0000 : CONNECT - PROXYOBJ: nil
-------
2018-05-14 13:25:36 +0000 : STARTING GET HTTP OBJ: nil
-------
2018-05-14 13:25:36 +0000 : CONNECT - POSTING: #<URI::HTTPS https://xxx.20.0.47/planner_register>
-------
2018-05-14 13:25:37 +0000 : CONNECT - POSTED: #<Net::HTTPOK 200 OK readbody=true>
-------
2018-05-14 13:25:37 +0000 : CONNECT - RESPONSE HASH: {"ipaddr"=>"xxx.20.0.47", "ipmask"=>"255.255.255.0", "node_version"=>"3.3.3.1", "ssh_pub_key"=> ...
```

- All Actors: When an unexpected response is received, a message will be displayed and a `response.txt` file is created.
- Network Actor, Protected Theater - Pull communication: `/opt/apps/verodin/node/log/verodin_registration`

DEBUG:10/07/2018 06:27:58 PM information.py:98: /var/run/dhclient-eth0.pid does not exist DHCP is false for eth0 DEBUG:10/07/2018 06:28:06 PM information.py:98: /var/run/dhclient-eth1.pid does not exist DHCP is false for eth1 DEBUG:10/09/2018 09:40:27 PM __init__.py:118: netifaces: eth0 does exist DEBUG:10/09/2018 09:40:27 PM __init__.py:118: netifaces: eth1 does exist DEBUG:10/09/2018 09:40:40 PM networking_centos.py:20: netifaces: eth0 does exist DEBUG:10/09/2018 09:40:40 PM information.py:98: /var/run/dhclient-eth0.pid does not exist DHCP is false for eth0 DEBUG:10/09/2018 09:40:40 PM networking_centos.py:20: netifaces: eth1 does exist DEBUG:10/09/2018 09:40:40 PM information.py:98: /var/run/dhclient-eth1.pid does not exist DHCP is false for eth1 INFO:10/09/2018 09:40:40 PM _system.py:207: Checking -backend.service for actor user INFO:10/09/2018 09:40:40 PM _system.py:230: Actor user: nodeone INFO:10/09/2018 09:40:40 PM _system.py:207: Checking -backend.service for actor user INFO:10/09/2018 09:40:40 PM _system.py:230: Actor user: nodeone DEBUG:10/09/2018 09:40:40 PM information.py:150: routing table is Kernel IP routing table Destination Gateway Genmask Flags Metric Ref Use Iface 0.0.0.0 10.10.20.1 0.0.0.0 UG 0 0 0 eth0 10.10.20.0 0.0.0.0 255.255.255.0 U 0 0 0 eth0 10.10.20.0 0.0.0.0 255.255.255.0 U 0 0 0 eth1 ...

- Network Actor, Protected Theater - Push communication: `/opt/apps/verodin/node/log/verodin_node_web`

DEBUG:05/14/2018 01:25:36 PM web_helpers.py:43: Received request for planner_registerDEBUG:05/14/2018 01:25:36 PM node_web.py:498: planner register DEBUG:05/14/2018 01:25:36 PM infrastructure.py:704: registering with planner DEBUG:05/14/2018 01:25:36 PM networking_centos.py:87: netifaces: ens32 does exist DEBUG:05/14/2018 01:25:36 PM web_helpers.py:51: planner_register returning result: {"ipaddr": "xxx.20.0.47", "ipmask": "255.255.255.0", "node_version": "3.3.3.1", "ssh_pub_key": "ssh-rsa ABGTR3NzaC1yc2EAAAADAQABAAACAQC1Xl+npYaXXunoiyqhj5NO5BkD4WtnzULiBXUifM4kYm4xjVpFkkLRA/6t8sJof/tvLl3uleAiAm85Dmq+S6xRyzfOs/q+uKS0N support@.com\n", "gateway": "172.20.0.1", "result": "success", "network_info": "{\"interfaces\": \"ens32: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500\\nine

- Windows Actor & Protected Actor:
  `C:\Program Files\Verodin\node\log\verodin_registration`

```
DEBUG:04/16/2018 06:22:50 PM vregister:332: calling discover_products with config sent from director
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:71: Starting discover_products function with 13 config entries
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:72: List of programs: {'updateassistant 1.12.0.0', 'update for windows 10 for x64-based systems
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:73: List of services: {'winmgmt windows management instrumentation', 'timebrokersvc time brok
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:77: Checking entry in config file: {'technology': {'tech_type': 'Antivirus', 'vendor': 'McAfee', 'produ
}
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:77: Checking entry in config file: ...
```

```
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:77: Checking entry in config file: ...
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:77: Checking entry in config file: ...
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:77: Checking entry in config file: ...
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:77: Checking entry in config file: ...
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:77: Checking entry in config file: ...
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:77: Checking entry in config file: ...
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:77: Checking entry in config file: ...
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:77: Checking entry in config file: ...
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:77: Checking entry in config file: ...
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:77: Checking entry in config file: ...
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:77: Checking entry in config file: ...
DEBUG:04/16/2018 06:22:55 PM product_discovery.py:113: Returning 0 found products: []
DEBUG:04/16/2018 06:22:55 PM vregister:338: making request to discovered_products with data: {'discovered_products': '[]', 'token': 'UAasZdMtUMd6r
DEBUG:04/16/2018 06:22:55 PM vregister:364: response from discovered_products: 200
```

Addressing Communication Issues during Registration

If you find logs are not updated as expected, the issue is generally related to communication. The following is a list of commands and their purpose.

> **TIP:** if you are in a cloud environment, the Validation Platform will use the local IP of the interface. In most cases involving a cloud provider, this is the internal private interface. If you Director can connect to this interface, no other config is needed. However, if the Director connects via a public ip assigned to that interface, you will need to **create a protected rule (https://docs.mandiant.com/home/protected-theater-configurations#Adding)** to allow communication.

- `vstatus` : Make sure all necessary services are running.
- `tcpdump` : Use `tcpdump -nei mgmt_interface` on both the Actor and Director, and then attempt registration (helps to have 2 terminal windows open); In Push mode, there should be packets leaving the Director, destined for the Actor; in Pull mode there should be packets leaving the Actor, destined for the Director
- `ifconfig -a` : Make sure interface assignments are accurate; check interface names, MAC addresses, and netmasks
- `netstat -nr` : Make sure routing tables are accurate; check interface names, default routes, and netmasks
- `netstat -natup` : Check that TCP port 443 is listening on both Director and Actor
- `traceroute` : To see the routes and interfaces being used for communication, you can use the traceroute command
- `curl` : Depending on Push vs Pull communication method, you should be able to get from Director to Actor `(curl -k https://*actor ip*/`), or Actor to Director (`curl -k https://*director ip*/`), using curl and targeting the management interface

> **IMPORTANT:** Ping is not a good tool to use in this instance. The Validation Platform drops ICMP on the management interfaces, so pinging them will result in no response. If you want to use ping, pair it with tcpdump so you see the traffic hitting the interface, but be aware you still won't get an ICMP ECHO response.

Restart Services

If the PT or Protect Actor services are down, you can restart them.

Restart the services from the command line

Run `vrestart` .

- Log into the Operating System and run the following:

```
C:\Program Files\Verodin\node\node\scripts\vrestart
```

Restart the services from the Director

> **NOTE:** This can only be completed if you are a power user or an admin.

1. Click **Environment** > **Protected Theaters**.
2. Locate the Protected Theater or Protected Actor you need, open its Action menu, and click **Edit**.
3. Click **Restart Services**.