Resolve Linux Connector SELinux Policy Fault

Contents

Introduction

This document describes the fault raised when the SELinux policy on the system prevents the connector from monitoring system activity.

Background information

The connector requires this rule to be in the Secure Enterprise Linux (SELinux) policy if SELinux is enabled and in enforcing mode:

```
allow unconfined_service_t self:bpf { map_create map_read map_write prog_load prog_run };
```

This rule is not present in the default SELinux policy on Red Hat-based systems. The connector attempts to add this rule through the installation of a SELinux policy Module named `cisco-secure-bpf` during an install or upgrade. The fault is raised if `cisco-secure-bpf` fails to install and load, or is disabled. The user is notified of a Fault 19 as described in the list of Cisco Secure Endpoint Linux Connector Faults if this fault is raised by the connector.

Applicability

This fault can be raised after a fresh install or upgrade of the Connector, or after modifying the SELinux policy of the system.

Operating systems

- Red Hat Enterprise Linux 7
- CentOS 7
- Oracle Linux (RHCK/UEK) 7

Connector versions

- Linux 1.22.0 and later

Resolution

There are two methods to resolve this fault:

1. Reinstall or upgrade the connector.
2. Manually modify the SELinux policy.

Install Dependency

Both methods require the "policycoreutils-python" package installed on the system to build and load the SELinux policy module. Run this command to install this package.

```
yum install policycoreutils-python
```

Reinstall or upgrade the connector

An SELinux policy Module named `cisco-secure-bpf` will be installed to provide the required SELinux policy modification during an install or upgrade of the connector. Perform a standard reinstall or upgrade of the connector for this resolution method.

Manually modify the SELinux policy

A system administrator must manually build and load a SELinux policy module to modify the SELinux policy. Perform these steps to load the required SELinux policy rule:

1. Save this in a file named `cisco-secure-bpf.te`

   ```
   module cisco-secure-bpf 1.0;
   require {
   type unconfined_service_t;
   class bpf { map_create map_read map_write prog_load prog_run };
   }
   #============= unconfined_service_t ==============
   allow unconfined_service_t self:bpf { map_create map_read map_write prog_load prog_run };
   ```

2. Build and load the module using these commands.

   ```
   checkmodule -M -m -o "cisco-secure-bpf.mod" "cisco-secure-bpf.te"
   semodule_package -o "cisco-secure-bpf.pp" -m "cisco-secure-bpf.mod"
   semodule -i "cisco-secure-bpf.pp"
   ```

3. Restart the Connector to clear the fault.

Verify the SELinux policy modification

Run this command to check if the `cisco-secure-bpf` SELinux policy module is installed.

```
semodule -l | grep cisco-secure-bpf
```

The SELinux policy modification has occurred if the output reports "`cisco-secure-bpf 1.0`".

Run this command to check if the required SELinux policy rule is present.

```
sesearch -A | grep "unconfined_t unconfined_t : bpf"
```

The fault clears after the connector is restarted if the output reports "`allow unconfined_service_t self:bpf { map_create map_read map_write prog_load prog_run };`".