OpenNMS Meridian Release Notes

OpenNMS Meridian v2019.1.0, Last updated 2019-11-05 20:02:09 EST

Table of Contents

OpenNMS Meridian Development Team

Tarus Balog <tarus@opennms.org>
David Hustace <david@opennms.org>
Benjamin Reed <ranger@opennms.org>

Please report any omissions or corrections to this document by creating an issue at http://issues.opennms.org.

OpenNMS Meridian 2019

System Requirements

- **Java 8 through 11**: OpenNMS Meridian 2019 runs on JDK 8 through 11. We recommend the most recent version of OpenJDK 11.
- **Default Heap Size**: The default heap size is now 2GB.
- **PostgreSQL 10 or higher**: Meridian 2019 requires any supported version of PostgreSQL 10 or higher.

What's New in Meridian 2019

Since Meridian 2018, we have introduced a large number of features, most notably Telemetryd (for processing streaming telemetry like NetFlow and sFlow), the Sentinel (for horizontal scaling of telemetry and other processing), and ALEC (for alarm correlation).

On top of that, there have been many other improvements and bug fixes since Meridian 2018.

Meridian 2019 roughly matches the feature set available in Horizon 25.

Architecture for Learning Enabled Correlation

Horizon 23 introduced support for correlation of alarms into meta-alarms called "situations" using an engine called the Architecture for Learning Enabled Correlation.

Situations are OpenNMS alarms that contain one or more triggering alarms, which allows them to be browsed, acknowledged, and unacknowledged just like any other alarm.

A high-level overview of the goal and implementation of correlation can be seen on the ALEC web site.

Changes to the Alarm Lifecycle

Alarm Clearing

Traditionally, OpenNMS has created and resolved alarms in pairs, with one alarm representing the triggering event (or events), and then a second alarm representing the resolution. Horizon 23 changes this default behavior to use a single alarm to track the problem state, incrementing the alarm count when it occurs while in a problem state, or when moving from resolved *back* into a problem state. Additionally, you can configure OpenNMS to create a new alarm if a problem happens again.

These behaviors are controlled by the introduction of 2 new settings in the `opennms.properties` file:

`org.opennms.alarmd.legacyAlarmState`
: This setting reverts to the old (pre-23) behavior of creating separate alarms for a problem and its resolution.

`org.opennms.alarmd.newIfClearedAlarmExists`

This setting forces Alarmd to create a new alarm if a problem reoccurs, rather than incrementing an existing alarm. (Note: this is ignored if `legacyAlarmState` is set to `true`.)

These improvements are covered in a lunch and learn video we published recently, if you would like to learn more.

Alarmd Architecture

To facilitate the implementation of ALEC, alarmd has been rearchitected to use Drools to manage the alarm lifecycle, rather than Vacuumd automations, triggers, and actions.

If are migrating changes to `vacuumd-configuration.xml` from an earlier Meridian release, it is strongly recommended you port them to the new Alarmd Drools context. The Drools rules are in the `$OPENNMS_HOME/etc/alarmd/drools-rules.d/` directory.

Additionally, we no longer generate `alarmCreated`, `alarmEscalated`, `alarmCleared`, `alarmUncleared`, `alarmUpdatedWithReducedEvent`, and `alarmDeleted` events. Instead, it is recommended that you add Drools rules to react to alarm changes.

For more complicated integrations, we also have a new API — `AlarmLifecycleListener` — for reacting to alarm changes.

Kafka Data Collection Sync

In addition to publishing events, alarms, and node inventory to Kafka, we now publish collected time-series data to the Kafka bus as well.

Sentinel

In addition to the Minion, we have added a new container-based subsystem called "Sentinel." The Sentinel is a Karaf container that can be configured to run a subset of OpenNMS daemons as a standalone tool, to aid in horizontal scaling and/or high availability.

Sentinel is designed to run our Karaf/Camel/SQS-based messaging bus, syslog listener, telemetry receiver, and Newts and Elasticsearch persistence.

Node and Interface Metadata

There is now support for associating arbitrary metadata with nodes and interfaces, including configuring arbitrary metadata in the requisition UI.

For details on using the metadata APIs, see the Admin Guide and the Developer Guide.

Elasticsearch 7.x Support

All of the features that leverage integrations with Elasticsearch i.e. event & alarm history, flows & situation feedback have been updated to support Elasticsearch 7.x. Elasticsearch versions before 7.x are no longer supported.

Given the pace of changes and the number of breaking changes between major versions of Elastisearch, we will focus on supporting a single major version of Elasticsearch per release moving forward.

Other Improvements

Since Meridian 2019 is based on Horizon 25, it contains all the fixes and updates that have occurred since Meridian 2018 was created from the Horizon 21 codebase.

For a more complete list of changes included in this release, see the "What's New" documentation for the following Horizon releases:

- Horizon 22
- Horizon 23
- Horizon 24
- Horizon 25

Release Meridian-2019.1.0

Release 2019.1.0 is the first release in the Meridian 2019 series.

The codename for 2019.1.0 is *Mercury*

Bug

- removed service will break BSM web ui (Issue NMS-9322)
- Event parameters no longer preserve ordering (Issue NMS-9827)
- The JMX-Cassandra service goes down for all the cluster when a single instance is down. (Issue NMS-10027)
- deleting a BSM monitor while an alarm is active doesn't clear the alarm (Issue NMS-10184)
- default event description is incorrect (Issue NMS-10346)
- Config tester doesn't detect missing xml datacollection file (Issue NMS-10396)
- BSM alarm severity is not being updated (Issue NMS-10578)
- snmp authentication error traps with Enhanced Linkd / bridge discovery (Issue NMS-10582)
- Zooming with Backshift is broken (Issue NMS-10635)
- Karaf shell history thrown out with bathwater on upgrade (Issue NMS-10664)
- Node detail page renders with no content when invalid node ID specified (Issue NMS-10679)
- Apparent memory leak in JMX collector, possibly restricted to "weird" JMX transports (Issue NMS-10684)
- Elasticsearch forwarding fails to recover after outage (Issue NMS-10697)
- Flow rest results for top N queries are not returned in the correct order (Issue NMS-12104)
- karaf.log appears on the root file system when running Minion/Sentinel on Ubuntu/Debian.

(Issue NMS-12125)

- WS-MAN doesn't work with JDK 11 (Issue NMS-12235)
- ReST API for meta-data doesn't support JSON (Issue NMS-12272)
- UI for meta-data is only present when using the horizontal layout (Issue NMS-12273)
- Groups disappear in classification UI (Issue NMS-12291)
- BSM simulation mode does not reset the last state (Issue NMS-12302)
- Web Assets Dependency Rollup 2019-09-24 (Issue NMS-12320)
- Memory leak in Drools engine for alarmd (Issue NMS-12322)
- Threshold state keys do not incorporate the collected resource's instance label (Issue NMS-12329)
- Reportd generated reports cause: "No bean named '' is defined" in Persisted Reports (Issue NMS-12337)
- InterfaceNodeCache doesn't remove deleted nodes immediately (Issue NMS-12338)
- Delivering a report with "-" in local part of email address is not working (Issue NMS-12342)
- Install guide for R-core is broken for CentOS 8 (Issue NMS-12352)
- Karaf feature install issue with opennms-core-tracing-jaeger (Issue NMS-12359)
- Fix requisition cache when accessing the Requisitions UI via "Edit in Requisition" (Issue NMS-12360)

Enhancement

- Refactor the compatibility matrix in the documentation (Issue NMS-9684)
- Be able to change the number of rows for the pagination control on the Requisitions UI (Issue NMS-9793)
- Documentation typo for /rest/ifservices on the developers guide (Issue NMS-9842)
- Remove alarm-change-notifier plugin (Issue NMS-10658)
- Add OpenTracing support for Camel (JMS) RPC (Issue NMS-10961)
- Support large buffer sizes in Kafka Sink Layer (Issue NMS-11126)
- Investigate OpenTracing for our RPC communications (Issue NMS-11177)
- RPC Metrics (Issue NMS-11517)
- Sink Metrics (Issue NMS-11540)
- Add a command to show configuration diffs (Issue NMS-12129)
- Add Web-Hook as delivery option (Issue NMS-12153)
- Add reply-to field to notification emails (Issue NMS-12224)
- Refactor Event Timestamps to ISO-8601 Format (Breaking Change) (Issue NMS-12263)
- Improve robustness of CassandraBlobStore for async operations (Issue NMS-12274)
- Clearing threshold states via shell should take effect immediately and not require restart (Issue

NMS-12277)

- BSM configuration breaks without being notifed (Issue NMS-12288)
- List Kafka RPC/Sink topics, Expose Metrics on Karaf shell (Issue NMS-12294)
- Create proper systemd files for OpenNMS, Minion and Sentinel (Issue NMS-12299)
- Add ability to update definitions when SNMP profile changes (Issue NMS-12307)
- Fix security vulnerability with jackson-databind (Issue NMS-12308)
- Availability boxes on node pages including sub pages differ (Issue NMS-12321)
- OpenNMS 25 Dependency Still Allows Old PostgreSQL Versions (Issue NMS-12341)
- Update base container image to use CentOS 8 (Issue NMS-12353)
- Remove floating OpenJDK dependencies in OCI build (Issue NMS-12354)
- Detect and help resolve Karaf bootstrap issues (Issue NMS-12356)
- Update CISCO-ENTITY-SENSOR-MIB threshold trap events to include alarm-data (Issue NMS-12362)
- switch core/web-assets from yarn to npm (Issue NMS-12363)
- Collect and display file descriptor statistics via JMX for OpenNMS and Minion (Issue NMS-12364)