## Table of Contents

What's New in OpenNMS Horizon 23 ................................................................. 2  
System Requirements .......................................................... 2  
Breaking Changes .............................................................. 2  
  Vacuumd Alarm Handling .................................................... 2  
New Features ................................................................. 2  
  OpenNMS Correlation Engine ("Sextant") Proof of Concept ................. 2  
  Alarmd Improvements ..................................................... 2  
  Kafka Data Collection Sync ............................................. 3  
  Sentinel ................................................................. 3  
Other Improvements ............................................................ 3  
Changelog ................................................................. 4  
  Release 23.0.0 .......................................................... 4
What’s New in OpenNMS Horizon 23

System Requirements

- **Java 8**: OpenNMS Horizon 23 requires Java 8 as the runtime environment. To run Horizon 23, we recommend the most recent version of Oracle JDK 8 for your platform.

- **PostgreSQL 9 or higher**: Horizon 23 requires Any supported version PostgreSQL 9 or higher. As of this writing, PostgreSQL 9.3 is the earliest version under active support, but OpenNMS is known to work with at least PostgreSQL 9.1 and up.

Breaking Changes

**Vacuumd Alarm Handling**

In previous OpenNMS releases, a large portion of the alarm workflow was handled by Vacuumd automations, triggers, and actions. As part of the work to implement alarm correlation, this logic has been moved to Drools, running inside Alarmd.

If you made any changes to `vacuumd-configuration.xml` related to alarms, 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.

Also, 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.

New Features

**OpenNMS Correlation Engine ("Sextant") Proof of Concept**

Horizon 23 introduces the groundwork for supporting correlation of alarms into meta-alarms called "situations" using an external engine called the OpenNMS Correlation Engine. 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 Sextant wiki page.

Please note that implementing OCE is an advanced topic and there is still work to do before it is ready for general use. While the OpenNMS core supports situations, the logic of creating them is part of the OCE, which is designed to run in a separate Karaf container. Full support for OCE and correlation is due to be released in Horizon 24.

**Alarmd Improvements**

In addition to the Alarmd Drools updates mentioned previously, there have been some other enhancements made to Alarmd.
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.

Kafka Data Collection Sync

In addition to publishing events, alarms, and node inventory to Kafka, we can 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 standalone 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.

This initial release of Sentinel is designed to run our Karaf/Camel/SQS-based messaging bus, syslog listener, Netflow receiver, and Newts and Elasticsearch persistence.

Other Improvements

• A number of monitors now support using placeholders in parameter values to allow runtime substitution of IP address, node ID, and so on. (See: NMS-10200)

• A new WSMan monitor has been added which can use WQL filters against an endpoint to detect arbitrary services.

• The web UI now fully supports configuring date formats and user time zone in opennms.properties which was partially introduced in Horizon 22.

• Daemon reloading can now be done from the Karaf shell. Also, Syslogd and Trapd can now respond to daemon reload events.

• The RadixTreeSyslogParser now handles Cisco Syslog Message formats.

• RPC to Minions is now fully supported when using Kafka rather than JMS.

• A bunch of tools for checking the health of OpenNMS, Minions, and Sentinel containers have
been added to the Karaf command-line. (Run `health:check --help` for more details.)

- Many other improvements were made to the flow support introduced in Horizon 22.
- Support for HTTP proxies has been added to a number of places where OpenNMS makes outgoing HTTP connections. ([NMS-10312], [NMS-10313])
- A number of VMware improvements have been made including configurable timeouts and handling of unacknowledged vSphere alarms.
- Many subsystems have been improved as part of the refactoring to make them capable of running in the Sentinel container.
- The OSGi Plugin Manager has been updated to version 1.1.0.
- A policy ([ScriptPolicy](https://example.com)) has been added to Provisiond that allows you to run arbitrary Groovy scripts before persistence. An example policy can be found in `$OPENNMS_HOME/etc/examples/script-policies/`.

## Changelog

### Release 23.0.0

OpenNMS Horizon 23.0.0 is the first release of the stable 23 series. It contains a number of bug fixes and enhancements, including support for alarm correlation, running OpenNMS components in an external container, and many other improvements. For a high-level overview of what's changed in OpenNMS 23, see [What's New in OpenNMS 23](https://example.com).

The codename for 23.0.0 is *Granny Weatherwax*.

### Bug

- VMWare-Center-Monitoring make for every virtual machine a login/logout (Issue [NMS-8204](https://example.com))
- The ReST API used to return XMLs with namespace, and now it doesn't (Issue [NMS-8524](https://example.com))
- Config-tester not validating varbind matching in event files (Issue [NMS-9821](https://example.com))
- enlinkd prints bridge forwarding table debug messages in output.log (Issue [NMS-10258](https://example.com))
- Document notifd's match-all parameter (Issue [NMS-10275](https://example.com))
- DefaultProvisionService logs noisily for monitored service having state "N" (Issue [NMS-10291](https://example.com))
- LSB init script does not start Minion on Ubuntu/Debian (Issue [NMS-10294](https://example.com))
- SNMP init script does not start Minion on Ubuntu/Debian (Issue [NMS-10294](https://example.com))
- Kafka monitor doesn't take service down (Issue [NMS-10308](https://example.com))
- Kafka consumers on Sentinel are not shown on kafka-manager (Issue [NMS-10310](https://example.com))
- Sentinel init scripts fail to start the Karaf instance (Issue [NMS-10330](https://example.com))
- navbar.ftl not rendering (Issue [NMS-10342](https://example.com))
- Wrong data type for certain Cassandra JMX counters (Issue [NMS-10352](https://example.com))
- Cannot override TTL when running the Karaf Command collections:collect through Minions (Issue [NMS-10367](https://example.com))
- Trendbox on start page is empty (Issue [NMS-10368](https://example.com))
• Erroneous INFO-level log messages during every forced node rescan (Issue NMS-10370)
• Wrong JMX MBeans for minions (Issue NMS-10372)
• Alarm processing is very slow when Kafka producer is enabled and Kafka is unavailable (Issue NMS-10378)
• webpack build runs even if source files are unchanged (Issue NMS-10381)
• Problems with featuresRepositories and featuresBoot in features.cfg on Sentinel (Issue NMS-10388)
• find-java.sh doesn’t understand newer JDK output (Issue NMS-10401)
• int overflow in InstallerDb causes bamboo failures (Issue NMS-10402)
• The configuration example for Elasticsearch on the admin-docs is incorrect (Issue NMS-10407)

Enhancement

• Merge User and Admin Guide (Issue NMS-10182)
• Allow use of placeholders in parameter values of monitors (Issue NMS-10200)
• Add support for persisting Data Collection to Kafka (Issue NMS-10215)
• apply centralized datetime rendering to freemarker template(s) (Issue NMS-10239)
• Update Syslogd to respond to reload configuration events (Issue NMS-10252)
• Bump asciidoctor from 1.5.6 to 1.5.7 (Issue NMS-10255)
• The Contribution file in our repository duplicates community guide (Issue NMS-10260)
• Only increment the alarm count if the alarm type is not a resolving event. (Issue NMS-10264)
• Remove non used core/doc module (Issue NMS-10266)
• Add support for Cisco Syslog Message formats to the RadixTreeSyslogParser (Issue NMS-10273)
• Update health:check command to check Kafka connectivity when no JMS involved (Issue NMS-10300)
• ServiceLookup and TelemetryAdapterRegistry don’t wait properly (Issue NMS-10303)
• Refactor timeseries applicationContext xml files with ‘timeseries’ prefix (Issue NMS-10304)
• Support reload daemon for Trapd (Issue NMS-10305)
• Create karaf shell commands for reload daemon (Issue NMS-10306)
• Be able to use Proxy for any Monitor or Collector that uses HttpClientWrapper directly (Issue NMS-10312)
• Be able to use Proxy for any Monitor or Collector that uses HttpClient via UrlFactory (Issue NMS-10313)
• Optionally create new alarms when a problem reoccurs (Issue NMS-10331)
• Implement a OSGi-aware CollectionAgentFactory (Issue NMS-10340)
• VmwareMonitor should go down for an entity with unacknowledged vSphere-Alarms (Issue NMS-10345)
• Link Node Page to Node Requisition (Issue NMS-10364)
• Improve concurrency in VMware Connection Pool (Issue NMS-10373)
• Add Configurable Timeouts to VMware vCenter Discovery (Issue NMS-10377)
• Set JVM flag to start Attach listener by default (Issue NMS-10384)
• Class paths for collectd selector strategy in documentation is not correct (Issue NMS-10387)