Documentation

SystemMonitoring

Basic mail interface to System Monitoring Suites. Also provides deeper integration to Nagios and Icinga2 (acknowledge on lock and check script). Version 6.0.11 Edition

Build Date:

2020-01-25

Table of Contents

Preface

This module implements a basic interface to System Monitoring Suites.

For Nagios, it works by receiving email messages sent by a Network Monitoring Suite. New tickets are created in case of component failures. Once a ticket has been opened messages regarding the effected component are attached to this ticket. When the component recovers, the ticket state can be changed or the ticket can be closed. If an open ticket for a given host/ service combination exists, all mails concerning this particular combination will be attached to the ticket until it's closed.

For Icinga2, it works by specifying a host and a service in ticket dynamic fields. This combination of host and service is used after a ticket lock is set to generate a HTTP request which is send to the configured Icinga2 host (please see configuration chapter). In Icinga2, this request is used to create or to confirm new incidents.

If you have questions regarding this document or if you need further information, please log in to our customer portal at portal.otrs.com with your OTRS ID and create a ticket.

You do not have an OTRS ID yet? Register here for free.

Chapter 1. Feature List

1. Control flow (Nagios)

The following diagram illustrates how mails are handled by this module and in which cases they trigger which action. Pretty much all checks are configurable using the regular expressions given by the parameters listed below.

```
Mail matches 'FromAddress'?
|
+-> NO  --> Continue with regular mail processing
|
+-> YES --> Does a ticket with matching Host/Service combination
            already exist in OTRS?
            |
            +-> NO  -> Does 'State:' match 'NewTicketRegExp'?
            |           |
            |           +-> NO  -> Stop processing this mail
            |           |             (silent drop)
            |           |
            |           +-> YES -> Create new ticket, record Host
            |                         and Service, attach mail
            |
            +-> YES -> Attach mail to ticket
                    -> Does 'State:' match 'CloseTicketRegExp'?
                        |
                        +-> NO  -> Continue with regular mail processing
                        |
                        +-> YES -> Change ticket type as configured in
                                      'CloseActionState'
```

Besides of a few additional sanity checks this is how the SystemMonitoring module treats incoming mails. By changing the regular expressions it should be possible to adopt it to different monitoring systems.

2. Icinga2 acknowledge

A new ticket is created with values in the specified dynamic fields which are needed as combination of host and service for the communication to the Icinga2 host. After this newly created ticket is locked to an agent, a HTTP request is send to the configured Icinga2 host. In the Icinga2 host a new acknowledge is created or an existing one is confirmed.

Chapter 2. System Requirements

1. Framework

The following OTRS Framework is required:

- 6.0.26

2. Packages

The following packages are required:

- None

3. Operating System

This package requires one of the following operating systems:

- None

4. Third Party Software

This third party software is required to use this package:

- A network monitoring system, such as Nagios, Icinga2, HP OpenView or similar, capable of sending out events via e-mail.

Chapter 3. Installation

The following instructions explain how to install the package. There are two possibilities.

1. Admin Interface

Please use the following URL to install the package utilizing the Admin Interface (please note that you need to be in the "admin" group).

http://localhost/otrs/index.pl?Action=AdminPackageManager

2. Command Line

Whenever you cannot use the Admin Interface for whatever reason, you may use the following command line tool ("bin/otrs.Console.pl Admin::Package::Install") instead.

```
shell> bin/otrs.Console.pl Admin::Package::Install /path/to/SystemMonitoring-6.0.11.opm
```

Chapter 4. Configuration

The package can be configured via the System Configuration in the Admin Interface. The following configuration options are available:

1. Icinga2::Acknowledge::Author

Navigation: Core::SystemMonitoring::Icinga2.

Icinga2 acknowledgement author.

2. Icinga2::Acknowledge::Comment

Navigation: Core::SystemMonitoring::Icinga2.

Icinga2 acknowledgement comment.

3. Icinga2::Acknowledge::Enabled

Navigation: Core::SystemMonitoring::Icinga2.

Icinga2 acknowledgement enabled?

4. Icinga2::Acknowledge::FreeField::Host

Navigation: Core::SystemMonitoring::Icinga2.

Name of the Dynamic Field for Host.

5. Icinga2::Acknowledge::FreeField::Service

Navigation: Core::SystemMonitoring::Icinga2.

Name of the Dynamic Field for Service.

6. Icinga2::Acknowledge::HTTP::Password

Navigation: Core::SystemMonitoring::Icinga2.

The HTTP acknowledge password.

7. Icinga2::Acknowledge::HTTP::URL

Navigation: Core::SystemMonitoring::Icinga2.

Icinga API URL.

8. Icinga2::Acknowledge::HTTP::User

Navigation: Core::SystemMonitoring::Icinga2.

The HTTP acknowledge user.

9. Icinga2::Acknowledge::Notify

Navigation: Core::SystemMonitoring::Icinga2.

Icinga2 acknowledgement notify.

10. Icinga2::Acknowledge::Sticky

Navigation: Core::SystemMonitoring::Icinga2.

Icinga2 acknowledgement sticky.

11. Nagios::Acknowledge::FreeField::Host

Navigation: Core::SystemMonitoring::Nagios.

Name of the Dynamic Field for Host.

12. Nagios::Acknowledge::FreeField::Service

Navigation: Core::SystemMonitoring::Nagios.

Name of the Dynamic Field for Service.

13. Nagios::Acknowledge::HTTP::Password

Navigation: Core::SystemMonitoring::Nagios.

The HTTP acknowledge password.

14. Nagios::Acknowledge::HTTP::URL

Navigation: Core::SystemMonitoring::Nagios.

The HTTP acknowledge URL.

15. Nagios::Acknowledge::HTTP::User

Navigation: Core::SystemMonitoring::Nagios.

The HTTP acknowledge user.

16. Nagios::Acknowledge::NamedPipe::CMD

Navigation: Core::SystemMonitoring::Nagios.

Named pipe acknowledge command.

17. Nagios::Acknowledge::NamedPipe::Host

Navigation: Core::SystemMonitoring::Nagios.

Named pipe acknowledge format for host.

18. Nagios::Acknowledge::NamedPipe::Service

Navigation: Core::SystemMonitoring::Nagios.

Named pipe acknowledge format for service.

19. Nagios::Acknowledge::Type

Navigation: Core::SystemMonitoring::Nagios.

Define Nagios acknowledge type.

20. PostMaster::PreFilterModule###00-SystemMonitoring

Navigation: Core::Email::PostMaster.

Basic mail interface to System Monitoring Suites. Use this block if the filter should run BEFORE PostMasterFilter.

21. PostMaster::PreFilterModule###1-SystemMonitoring

Navigation: Core::Email::PostMaster.

Basic mail interface to System Monitoring Suites. Use this block if the filter should run AFTER PostMasterFilter.

22. SystemMonitoring::LinkTicketWithCI

Navigation: Core::SystemMonitoring.

Link an already opened incident ticket with the affected CI. This is only possible when a subsequent system monitoring email arrives.

23. SystemMonitoring::SetIncidentState

Navigation: Core::SystemMonitoring.

Set the incident state of a CI automatically when a system monitoring email arrives.

24. Ticket::EventModulePost###9-Icinga2Acknowledge

Navigation: Core::Event::Icinga2.

Ticket event module to send an acknowledge to Icinga2.

25. Ticket::EventModulePost###9-NagiosAcknowledge

Navigation: Core::Event::Nagios.

Ticket event module to send an acknowledge to Nagios.

Chapter 5. File List

This list shows all included files and the referring permissions.

- (660) doc/en/SystemMonitoring.xml

- (660) Kernel/Config/Files/XML/SystemMonitoring.xml

- (660) Kernel/Config/NagiosCheck.pm.example

- (660) Kernel/Language/bg_SystemMonitoring.pm

- (660) Kernel/Language/de_SystemMonitoring.pm

- (660) Kernel/Language/es_MX_SystemMonitoring.pm

- (660) Kernel/Language/es_SystemMonitoring.pm

- (660) Kernel/Language/hu_SystemMonitoring.pm

- (660) Kernel/Language/it_SystemMonitoring.pm

- (660) Kernel/Language/ja_SystemMonitoring.pm

- (660) Kernel/Language/pl_SystemMonitoring.pm

- (660) Kernel/Language/pt_BR_SystemMonitoring.pm

- (660) Kernel/Language/ru_SystemMonitoring.pm

- (660) Kernel/Language/sk_SK_SystemMonitoring.pm

- (660) Kernel/Language/sr_Cyrl_SystemMonitoring.pm

- (660) Kernel/Language/sr_Latn_SystemMonitoring.pm

- (660) Kernel/Language/zh_CN_SystemMonitoring.pm

- (660) Kernel/System/Console/Command/Maint/SystemMonitoring/NagiosCheck.pm

- (660) Kernel/System/Console/Command/Maint/SystemMonitoring/NagiosCheckTicket-Count.pm

- (660) Kernel/System/PostMaster/Filter/SystemMonitoring.pm

- (660) Kernel/System/Ticket/Event/NagiosAcknowledge.pm

- (660) Kernel/System/Ticket/Event/Icinga2Acknowledge.pm

- (660) scripts/test/Console/Command/Maint/SystemMonitoring/NagiosCheck.t

- (660) scripts/test/Console/Command/Maint/SystemMonitoring/NagiosCheckTicketCount.t

- (660) scripts/test/SystemMonitoring.t

- (660) scripts/test/sample/NagiosCheckTesting.pm

- (660) scripts/test/sample/SystemMonitoring1.box

- (660) scripts/test/sample/SystemMonitoring2.box

- (660) var/packagesetup/SystemMonitoring.pm

Chapter 6. Tests

This module has been quality assured according to the current OTRS guidelines.

1. Test Cases

To test this package please follow the examples described in the Usage section.

2. Unit Test

To ensure the quality of the module, several so-called unit tests were created, to test the functionalities of this module. These unit tests can be run via command line.

ATTENTION: Please never run unit tests on a productive system, since the added test data to the system will no longer be removed. Always use a test system.

Run the package specific unit tests

To run only the unit test which will be delivered with this package, use the following command on the command line:

```
shell> perl bin/otrs.Console.pl Dev::UnitTest::Run --test Console/Command/Maint/
SystemMonitoring/NagiosCheck
shell> perl bin/otrs.Console.pl Dev::UnitTest::Run --test Console/Command/Maint/
SystemMonitoring/NagiosCheckTicketCount
shell> perl bin/otrs.Console.pl Dev::UnitTest::Run --test SystemMonitoring
```

Run all available unit tests

To run all available unit tests, use the following command on the command line:

```
shell> perl bin/otrs.Console.pl Dev::UnitTest::Run
```

Chapter 7. Change Log

6.0.11 / 2020-01-25 03:04:13

- - Updated minimum Framework version. - Updated translations. - Updated copyright.

6.0.10 / 2019-12-14 01:43:46

- - Updated minimum required version. - Updated copyright.

6.0.9 / 2019-07-01 07:08:26

- - Updated minimum framework version to OTRS 6.0.20. - Updated translations.

6.0.8 / 2019-05-18 01:50:41

- - Added minimum framework version.

6.0.7 / 2019-04-10 20:03:15

- - Updated translations.

6.0.6 / 2019-01-15 02:51:45

- - Updated copyright.

6.0.5 / 2018-12-11 06:10:35

- - Tidied.

6.0.4 / 2018-09-15 05:27:31

- Updated license information.

6.0.3 / 2018-07-26 02:36:30

- - Updated translations, thanks to all translators. - Fixed bug#13679 - Icinga2 support is not displayed in package description. - Added Hungarian documentation. - Updated documentation.

6.0.2 / 2018-03-07 00:53:57

- - Fixed: OTRS dies if a system monitoring mail with an empty body will be received (bug#9012). - Updated copyright.

6.0.1 / 2017-11-15 01:52:10

- - Updated config chapter on documentation. - Updated translations, thanks to all translators. - Removed not needed StatsCleanUp call.

5.0.92 / 2017-11-08 18:32:05

- - Added support for Icinga2.

5.0.91 / 2017-10-17 04:02:26

- - Improved the communication log for the packagesetup functions. - Updated translations, thanks to all translators. - Ported to OTRS 6.