opEvents - Solution Guide - Setup Email Notifications and Other Actions

Feature Overview

opEvents has the ability to notify people via email and conduct other actions based on criteria the administrator selects.

Order of operation:

- An event matches criteria defined in an action rule
- The event action policy indicates that an email action is to be taken, with a given 'contact'.
- Or: the event action activates an escalation policy, the event is escalated after some time and the Escalation policy indicates an email action to take, again with a given 'contact.
- The named 'contact' is resolved to one or more email addresses.
- Emails are sent via the configured email server.

Related Wiki Articles

- Event Actions and Escalation (opEvents 4)
- Email templates in opEvents
- NMIS8 Contacts

Configuration

Define an Email Server

opEvents requires an email server in order to send notifications via email. This email server is configured in `/usr/local/omk/conf/opCommon.nmis`.

opCommon.nmis

```
   'email' => {
    'mail_domain' => 'yourdomain.com',
    'mail_from' => 'yourmailname@yourdomain.com',
    'mail_password' => 'your_password',
    'mail_server' => 'smtp.yourdomain.com',
    'mail_server_port' => 25,
    'mail_use_tls' => 'true',
    'mail_user' => 'your_user_account@your_domain.com'
  },
```

Open `opCommon.nmis` in an editor, find the email section and update the values to match the email server that opEvents should utilise.

Here is a brief description of the fields:

Field Name	Description
mail_domain	This is required for some mail servers that enforce strict HELO messages. Using your company domain here is a good idea.
mail_from	This is the From address for email notifications. Emails will go out with this as the sender.
mail_user	This is the mail user name for authenticating at your mail server. Leave this blank if you don't need to authenticate at your mail server.
mail_password	This is the password for authenticating at your mail server. Leave this blank if you don't need to authenticate at your mail server.

mail_server	The FQDN (or IP address) of your outgoing mail server.
mail_server_port	The port number your mail server listens on for SMTP conversations. Common choices are 25 and 587, but note that 587 commonly requires authentication!
mail_use_tls	If you select true here, then opEvents will try to negotiate STARTTLS encryption with your mail server. Not useful if your mail server is `localhost`.
mail_use_sasl	disregard: deprecated, should not be present in new installations.

Optional - Create an Email Template

It's possible to customise the email format by updating /usr/local/omk/conf/EventEmails.nmis. This article explains how to do this: Email templates in opEvents

Configure a Contact

A 'contact' is a variable that can represent one or many email addresses. opEvents can utilize the NMIS contacts file (/usr/local/nmis8/conf/Contact.nmis) or the OMK contacts (/usr/local/omk/conf/COntacts.nmis) file. In order to determine which file your system is using look in opCommon.nmis; find the opevents section then look for the **opevents_contacts** attribute.

opCommon.nmis

```
  'opevents' => {
    'black_list_enabled' => 'true',
    'log_archive_enabled' => 'true',
    'nmis_non_stateful_events' => 'Node Configuration Change|Node Reset',
    'opevents_action_max_runtime' => 30,
    'opevents_action_policy' => '<omk_conf>/EventActions.nmis',
    'opevents_application_heading' => undef,
    'opevents_archivelogs_purge_older_than' => undef,
    'opevents_auto_acknowledge' => 'true',
    'opevents_auto_acknowledge_up' => 'true',
    'opevents_auto_create_nodes' => 'true',
    'opevents_contacts' => '<omk_conf>/Contacts.nmis',
```

In the example above (last line) the OMK contacts file is being used.

In Contacts.nmis create a new contact or modify and existing one with the desired email address(es).   If desired an email template may be associated with the contact. Multiple emails may be assigned to the Email attribute, simply separate them by a comma.

Contacts.nmis

```
%hash = (
  'contact1' => {
    'Contact' => 'Contact1',
    'Phone' => '',
    'DutyTime' => '06:20:MonTueWedThuFri',
    'Level' => '(Fatal|Critical|Major|Minor|Warning|Normal)',
    'TimeZone' => '0',
    'Pager' => '',
    'Mobile' => '5555551234',
    'Location' => 'default',
    'Email' => 'nobody@localhost',
    'EmailTemplate' => 'default',
  }
);
```

For more information about how to configure Contacts.nmis for use by NMIS see this page: NMIS8 Contacts

Create an Escalation Policy

Escalation policies are created in /usr/local/omk/conf/EventActions.nmis. Find the 'escalate' section and add or modify an existing policy. This is an example of a new escalation policy with the minimum required information.

EventActions.nmis

```
         'escalate' => {
                'exampleEmailPolicy' => {
                  'name' => 'exampleEmailPolicy',
                  'IF' => {
                        priority => '>= 0',
                        },
                        '10' => 'email(contact1)',
                },
```

This example states that 10 seconds after the policy is invoked it will email addresses associated with 'contact1' if the priority of the event is greater than zero.

For more information about creating Escalation policies see this page: Event Actions and Escalation (opEvents 4)

Create an Action

An action is required that will invoke the escalation policy. Actions are created in the 'policy' section of EventActions.nmis. The opEvents administrator will determine where best to place the rule with EventActions.nmis. Here is one example.

EventActions.nmis

```
                                 '40' => {
                                        IF => 'event.priority >= 5',
                                        THEN => 'escalate.exampleEmailPolicy()',
                                        BREAK => 'false'
                                },
```

In this example any event with a priority greater than or equal to 5 will fire the escalation policy 'exampleEmailPolicy. Because the BREAK attribute is set to false, any rules following this one in the same hierarchy will continue to be processed.

For more information about Action Policies see this page: Event Actions and Escalation (opEvents 4)

Verification

The opevents-cli.exe utility found in /usr/local/omk/bin may be used to create a test event. Create an event that will match the previously configured action rule.

```
[root@opmantek conf]# /usr/local/omk/bin/opevents-cli.exe act=create-event node=testNode4 stateful=node
state=down event="Node DOWN" priority=5
5997bb8cce2c2e6d9453c101
[root@opmantek conf]#
```

This command will return an event ID. Go to the event context page for this event. In the Actions section of the page there should be an entry stating an email was successfully sent. Keep in mind that this entry will not be present until the flap window and escalation timers have elapsed.