

Custom Notification Methods for NMIS Events

- [Prerequisites](#)
- [Introduction](#)
- [Notification Extensibility in NMIS](#)
 - [What are Notifications in NMIS](#)
- [Adding new Notification Types to NMIS](#)
 - [Creating a New Notification Method Package](#)
 - [Add a New Notification to Escalations.nmis](#)
 - [Creating Your Own Notification](#)
 - [Sample Contact Details](#)
 - [Sample NMIS Event Details](#)
- [Feedback](#)

Prerequisites

NMIS version 8.3.24G or greater.

Unix Shell access to the NMIS server and suitable Unix privileges to edit the NMIS configuration files, usually a member of the group "nmis" or the root user.

Introduction

While working with customers who wanted to extend NMIS and use various methods for notifications, making it a highly extensible Network Management System. To better support notification methods, the logic to initiate a notification and the method for notification has been split, so that additional notification methods can be added without having to modify the core part of NMIS.

This article will briefly describe how this capability works and how it supports operational agility.

Notification Extensibility in NMIS

What are Notifications in NMIS

NMIS raises an event internally and then applies an escalation policy to the event, the result of this escalation policy is a notification. A notification method can be logging, email, SMS, netsend, any number of types. In the past you needed to modify the core NMIS code to add a notification method, but now you can write a notification method outside of NMIS core code and it will be used as required at runtime.

Adding new Notification Types to NMIS

Creating a New Notification Method Package

To add a new notification method, you will need to copy the example Perl code which is in the distribution, which should be at `/usr/local/nmis8/lib/Notify/mylog.pm`, this is a functional notification method. If you wanted to add something called `netsms` you would copy that file to `netsms.pm`, so

```
$ cp /usr/local/nmis8/lib/Notify/mylog.pm /usr/local/nmis8/lib/Notify/netsms.pm
```

Then you need to edit the file `netsms.pm` and change the following

```
package Notify::mylog;
```

To be the package name for `netsms`, e.g.

```
package Notify::netsms;
```

At this point you can run this at test it, you can do this by adding an escalation type, which will be `netsms`.

Note, NMIS8 will lowercase the escalation data, so all methods for escalations need to be in lower case.

Add a New Notification to Escalations.nmis

Looking at Escalations.nmis which is in the conf directory at /usr/local/nmis8/conf/Escalations.nmis, look for the entry "default_default_default_default__"

```
'default_default_default_default__' => {
  'Event' => 'default',
  'Event_Element' => '',
  'Event_Node' => '',
  'Group' => 'default',
  'Level0' => 'syslog:localhost,json:localhost,netsms:Contact1',
  'Level1' => '',
  'Level10' => '',
  'Level2' => '',
  'Level3' => '',
  'Level4' => '',
  'Level5' => '',
  'Level6' => '',
  'Level7' => '',
  'Level8' => '',
  'Level9' => '',
  'Role' => 'default',
  'Type' => 'default',
  'UpNotify' => 'true'
},
```

Modify the line for "Level0" and add ",netsms:Contact1", you will now get notifications using the code in /usr/local/nmis8/lib/Notify/netsms.pm, the next time you get a notification look in /tmp/mylog for the file.

Creating Your Own Notification

Now you know its working you can modify the contents of the subroutine "sendNotification" to do what you want to do, in the example it is just logging to a file, but you can put anything here Perl can do (which is alot).

You have access to the following data.

Sample Contact Details

Field	Description
Contact	The contact ID or name
DutyTime	The on duty string for this contact
Email	The email address for this contact
Location	The location of this contact
Mobile	Mobile number can be used for SMS alerting
Pager	Pager value (does anyone still have these?)
Phone	The phone number
TimeZone	Timezone offset from GMT

Sample NMIS Event Details

Depending on your system and what features you have enabled, there may be more values, but at a minimum you will have the following

Field	Description
ack	Has the event been acknowledged or not?
details	The details associated with the event
element	The element associated with the event, e.g. interface name

escalate	What level is the escalation at
level	Criticality of the event
node	The node the event is related to.
time	The time the event was generated
notify	Who was notified about the event.

Feedback

We would love you get your feedback, please let us know if you had any problems or would like more information at contact@opmantek.com