

NMIS Events sent to IBM Tivoli Netcool using syslog

Table of contents

- [Summary](#)
- [Requirements](#)
- [Solution](#)
 - [Installation](#)
 - [Update NMIS escalations](#)
 - [Update NMIS contacts](#)
 - [Wait for events to be generated](#)
 - [Local Syslog Testing](#)

Summary

Have certain NMIS events sent to IBM Tivoli Netcool using syslog.

Requirements

Send certain events to Netcool, which can be done by ignoring the events not required.

Solution

An NMIS notification plugin was created to send all events, except those in the ignore list to the configured syslog server.

Installation

The code for this solution is included in the NMIS9 contrib folder which is available in the installation or from [NMIS9@GitHub](#), it will be in the folder nmis9/contrib/noc_netcool_syslog

A good option to install is to create a util folder e.g. /usr/local/nmis9/util and then create a symbolic link so the file will run with the correct paths, and copy the nocSyslog.nmis to the NMIS9 conf folder

Some handy commands (sudo might be required):

```
sudo cp nocSyslog.nmis nocIgnoreList.txt /usr/local/nmis9/conf
sudo cp noc.pm /usr/local/nmis9/lib/Notify
sudo /usr/local/nmis9/bin/nmis-cli act=fixperms
sudo service nmis9d restart
```

Update nocSyslog.nmis with your needed config, the options should be self explanatory except for extra_logging, which if enabled will give you some logging to nmis.log with when events are sent over syslog to NetCool.

```
%hash = (
  'syslog' => {
    'syslog_facility' => 'local3',
    'syslog_server' => 'localhost:udp:514',
    'extra_logging' => 1,
  }
);
```

Update NMIS escalations

To configure this, you add the method in the policy where you require it, for example

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

Update NMIS contacts

Here "noc:noc_contact" has been added to the escalation policy. The first part noc names the custom notification method matching the code in [noc.pm](#) and noc_contact must be a contact in the Contacts.nmis file in lower case, this is used for criticality filtering and duty time only.

```
'noc_contact' =>
{ 'Contact' => 'noc_contact', 'DutyTime' => '00:24:MonTueWedThuFriSatSun', 'Email' => 'nobody@localhost',
'Location' => 'default', 'Level' => '(Fatal|Critical|Major|Minor|Warning|Normal)', 'Mobile' => '', 'Pager' =>
'', 'Phone' => '', 'TimeZone' => '0' }
,
```

Wait for events to be generated

Wait some NMIS events to be generated or create some conditions to generate the events, e.g. mess with the routes.

You should find your NMIS events in the remote syslog server.

Local Syslog Testing

To test locally, add the following to /etc/rsyslog.conf for testing and restart rsyslogd.

```
# provides UDP syslog reception
module(load="imudp")
input(type="imudp" port="514")

# provides TCP syslog reception
module(load="imtcp")
input(type="imtcp" port="514")

local3.*                /usr/local/nmis9/logs/noc.log
```