NMIS Model Policy

Introduction

NMIS 8.6 brings in a new mechanism for adjusting a model's behaviour for particular nodes: the Model Policy system. In version 8.6.0G it allows you to specify flexible rules for adding or removing systemHealth model sections for specific nodes (or groups of nodes). Future releases will extend the model policy to also cover things like threshold values and the like

The Model Policy Document

The installer will install a default model policy document in `conf/Model-Policy.nmis`. The original/default file will also remain available in the `install` directory, and contains helpful comments.

The structure of the policy document is quite simple but fairly flexible:

- The policy consists of any number of rules.
  The rules are evaluated in order of their numeric key; fractional numbers are supported to simplify insertion.
  Only the first matching rule is applied.
- Each rule *must* express what changes to systemHealth should be made.
  The changes are expressed as a list of systemHealth section name plus the desired activation state (true or false).
  systemHealth sections not named are not modified and therefor default to True or on (so check the default section for what would otherwise be disabled).
- Each rule *may* include any number of filter expressions, which determine whether the rule should be applied to a particular node.
  All given filter expressions must match simultaneously for the rule to be considered a match.
  A filter expression defines a node property or configuration setting to be compared against an explicit list of acceptable values, or a regular expression.
  Node properties are given as `node.<propname>`, and for configuration settings you'd use `conf.<configsetting>`.
- All configuration settings are available, using the prefix `conf.` and the same names as seen in `conf/Config.nmis`.
- The available node properties are: the static ones from the node configuration, plus the more dynamic ones from the `system` section in the node's "node info" file (`var/<nodename>-node.json`).
- A Model Policy document *may* also include an extra section named `_display`, which controls in what order the default policy's entries should be shown in the Configuration GUI.
  See the default policy for an example.

It should be noted that as Only the first matching rule is applied and therefor the default rule is not subsequently applied you should include all the relevant "false" sections from the default rules into your rule. For example if you wanted to turn on just one mpls system health section you would set that as true in your rule and you would also include all the other "false" lines which are relevant to your model in the rule.

Example Policy

Here is a partial example policy:

```
%hash = (
        # rule numbers may be fractional numbers (for easy insertion)
        # first matching rule terminates the policy application
        10 => {
            # filter keys: node.xyz or config.abc; node.nodeModel is the (possibly dynamic) current model
            # filter values: string, list of strings,
            # or regexp (=string with //, optional case-insensitive //i)
            IF => { 'node.name' => ['node1','node2'],
                    'node.location' => '/def.*/',
                    'config.auth_ldap_server' => '/192\./', },
            # sections to adjust, only systemHealth supported so far
            systemHealth => {
                'fanStatus' => 'true',    # add if not present
                'tempStatus' => 'false', # remove if present
            },
        },
        20 => {
            IF => { 'node.name' => 'embedded' },
            systemHealth => {
                diskIOTable => 'false' # this node runs off r/o flash disk
            },
        },
        999 => {    # the fallback/defaults, without filter
            systemHealth => {
                cdp => 'true',
                lldp => 'true',
                bgpPeer => 'true',
                ospfNbr => 'true', }
        } );
```

The first rule applies to at most two particular nodes (because of the given list of node.name values), and only if their location property starts with "def" and only if the NMIS configuration is set up for an LDAP server in the 192.0.0.0/8 network. For all systems that match these restrictions the fanStatus and tempStatus model sections are enabled.

The second rule disables the diskIOTable model section for a specific system that doesn't have real 'disks', just a readonly flash drive.

The last rule does not have any filtering IF clauses, therefore applies to all nodes and thus it serves to set the "default" policy. As mentioned above only the first matching rule is applied, hence the default rule will only apply to nodes where rules 10 and 20 have not matched.

Configuration GUI

As of version 8.6.0G, the Model Policy configuration page (under Setup -> Model Policy) lets you adjust only the default policy (ie. the highest numbered policy without filters).

For each of the known systemHealth section names you are given the option to enable or disable it.