NAME

Munin::Node::Configure::Plugin – Class representing a plugin, along with its installed and suggested services.

SYNOPSIS

```
my $plugin = Munin::Node::Configure::Plugin->new();
```

METHODS

new(%args)

Constructor.

Required arguments are 'name' and 'path', which should be the basename and full path of the plugin, respectively.

is_wildcard()

Returns true if the plugin is a wildcard. In the case of SNMP plugins, only double-wild plugins will return true (ie. 'snmp__memory' would return false, but 'snmp__if_' would return true).

is_snmp()

Returns true if the plugin is an SNMP plugin.

in_family(@families)

Returns true if plugin's family is in `@families`, false otherwise.

is_installed()

Returns 'yes' if one or more links to this plugin exist in the service directory, 'no' otherwise.

suggestion_string()

Returns a string detailing whether or not autoconf considers that the plugin should be installed. The string may also report the reason why the plugin declined to be installed, or the list of suggestions it provided, if this information is available.

installed_services_string()

Returns a string detailing which wildcards are installed for this plugin.

services_to_add()
services_to_remove()

Return a list of service names that should be added or removed for this plugin.

add_instance($name)

Associates a link from the service directory with this plugin.

add_suggestions(@suggestions)

Adds `@suggestions` to the list of suggested wildcards for this plugin. They are not validated.

read_magic_markers()

Sets the family and capabilities from the magic markers embedded in the plugin's executable, as specified by <http://munin−monitoring.org/wiki/ConcisePlugins#Magicmarkers>

parse_autoconf_response(@response)

Parses and validates the autoconf response from the plugin, in the format specified by <http://munin−monitoring.org/wiki/ConcisePlugins#autoconf>

Invalid input will cause an error to be logged against the plugin.

parse_suggest_response(@response)

Validates the suggestions from the plugin.

Invalid suggestions will cause an error to be logged against the plugin.

parse_snmpconf_response(@response)

Parses and validates the snmpconf response from the plugin, in the format specified by <http://munin−monitoring.org/wiki/ConcisePlugins#suggest>

Invalid or inconsistent input will cause an error to be logged against the plugin.

log_error($message)

Logs an error for later retrieval. The error will also be displayed if debugging output is enabled.