



# **ScalixReady**

*Technical Note*

## **Nagios in a Scalix Environment**

23 November 2004  
Version: 1.0

# Table of Contents

<b>1.</b>	<b>Revision History</b> .....	<b>3</b>
<b>2.</b>	<b>Introduction</b> .....	<b>3</b>
<b>3.</b>	<b>Versions and Prerequisites</b> .....	<b>3</b>
<b>4.</b>	<b>Installation</b> .....	<b>3</b>
4.1.	Nagios.....	3
4.2.	Scalix Plugins for Nagios .....	3
<b>5.</b>	<b>Configuration</b> .....	<b>4</b>
5.1.	Nagios.....	4
5.2.	Scalix Plugins for Nagios - Local.....	4
5.2.1.	nagios.cfg.....	4
5.2.2.	Host definitions.....	4
5.2.3.	Service Definitions .....	5
5.2.4.	Contact Definitions .....	5
5.2.5.	Host Group Definitions.....	6
5.2.6.	Contact Group Definitions.....	6
5.2.7.	Command Definitions.....	6
5.2.8.	Escalation Definitions.....	7
5.3.	Scalix Plugins for Nagios – Remote (nrpe).....	7
5.3.1.	nrpe.cfg.....	7

## 1. Revision History

Version	Date	Authors	Comments
1.0.0	11/19/2004	RK	Initial version

## 2. Introduction

Nagios is an Open Source monitoring package available for RedHat and SuSE Linux . Scalix provides plugins for Nagios so that the various services, queues, logs and disk usage statistics can be monitored from the Nagios console. The instructions below will configure the plugins to work with the Nagios system.

## 3. Versions and Prerequisites

Redhat Linux EL 3.2.3-34  
 Scalix Server 9.1.0.81  
 nagios-1.2.0  
 nagios-plugins-1.3.1-10  
 nagios-nrpe-2.0-1  
 nagios-plugins-nrpe-2.0.3

## 4. Installation

### 4.1. Nagios

Download and install the Nagios rpms.

```
rpm -iv nagios-1.2.0.platform.i386.rpm
rpm -iv nagios-plugins-1.3.1.10.platform.i386.rpm
```

If a remote Scalix server will also be monitored, you will need to download and install the Nagios nrpe modules on the remote servers.

```
rpm -iv nagios-1.2.0.platform.i386.rpm
rpm -iv nagios-plugins-1.3.1.10.platform.i386.rpm
rpm -iv nagios-plugins-nrpe-2.0.3.platform.i386.rpm
rpm -iv nagios-nrpe-2.0-1.platform.i386.rpm
```

Once the rpms have been installed, a new user and group called nagios will have been created. The nagios user must be added to the /etc/sudoers file using the visudo editor. This holds true for both the local and remote Nagios installations. The entry for nagios should look like:

```
nagios ALL = NOPASSWD: /usr/lib/nagios/plugins/check_queues.py,
/usr/lib/nagios/plugins/check_daemons.py, /usr/lib/nagios/plugins/check_services.py
```

### 4.2. Scalix Plugins for Nagios

Once Nagios is installed and working, unpack the Scalix Plugins for Nagios as follows:

```
tar xvzf scalix-nagios.tar.gz
```

This will create a scalix-nagios subdirectory that contains a scalix/etc, a scalix/libexec and a scalix/nrpe subdirectory. Create a subdirectory called /etc/nagios/scalix and copy the contents of the scalix-nagios/scalix/etc subdirectory to this subdirectory and change the ownership of the files to the nagios user.

```
mkdir /etc/nagios/scalix
cp scalix-nagios/scalix/etc/* /etc/nagios/scalix
chown nagios /etc/nagios/scalix/*
```

Copy the contents of the scalix-nagios/scalix/libexec subdirectory to the nagios plugin subdirectory:

```
cp scalix-nagios/scalix/libexec/* /usr/lib/nagios/plugins
```

If a remote Scalix server will also be monitored, you will need to install Nagios, the Nagios nrpe RPMs and the Scalix Plugins for Nagios on the remote host. You can then use the nrpe.cfg file in the scalix-nagios/scalix/nrpe subdirectory as the basis for your remote servers nrpe.cfg file. If you're already monitoring the remote system with Nagios, see the configuration section for details on modifying your existing nrpe.cfg file. You will also need to copy the contents of the the scalix-nagios/scalix/libexec to the remote hosts nagios plugin subdirectory:

```
scp scalix-nagios/scalix/libexec/* root@REMOTEHOST:/usr/lib/nagios/plugins
```

## 5. Configuration

### 5.1. Nagios

The Nagios configuration is beyond the scope of this document. However, once the Nagios software has been installed and configured to work with the Apache HTTP server, the system itself provides a great deal of online help.

### 5.2. Scalix Plugins for Nagios - Local

The Scalix Plugins for Nagios come with the following configuration files:

- hosts.cfg – Defines your Scalix Servers
- services.cfg – Identifies the services running on your Scalix Server
- contacts.cfg – Identifies the person/people who are to be contacted when a problem occurs
- hostgroups.cfg – A grouping of one or more hosts for the purpose of simplifying notifications
- contactgroups.cfg – A grouping of one or more contacts for the purpose of simplifying notifications
- checkcommands.cfg – Command definitions
- escalations.cfg – Optional method to escalate notifications for a particular Scalix server

#### 5.2.1. nagios.cfg

In order for Nagios to “see” the Scalix plugin configuration files, the /etc/nagios/nagios.cfg file must be edited and the following lines added:

```
#Scalix Plugin Configuration Files
cfg_file=/etc/nagios/scalix/checkcommands.cfg
cfg_file=/etc/nagios/scalix/contactgroups.cfg
cfg_file=/etc/nagios/scalix/contacts.cfg
cfg_file=/etc/nagios/scalix/escalations.cfg
cfg_file=/etc/nagios/scalix/hostgroups.cfg
cfg_file=/etc/nagios/scalix/hosts.cfg
cfg_file=/etc/nagios/scalix/services.cfg
```

#### 5.2.2. Host definitions

The hosts.cfg file contains the names and IP addresses of your Scalix servers. An example entry would look something like this:

```
# Scalix Server 1 host definition
define host{
    use                generic-host          ; Name of host template to use

    host_name         Scalix-Server-01
    alias              Scalix Server #1
    address            10.1.5.79
    check_command      check-host-alive
    max_check_attempts 10
```

```

notification_interval 120
notification_period 24x7
notification_options d,u,r
}

```

Where `host_name` is not necessarily the servers actual hostname (although it could be), but rather the name by which Nagios identifies the server. In the example above, the host is named `Scalix-Server-01` and its IP address is `10.1.5.79`. If you choose to change `host_name` field, you **must** also change it in the `hostgroups.cfg` and `services.cfg` files (see below). The IP address must be changed to your Scalix server's IP address.

### 5.2.3. Service Definitions

The `services.cfg` file contains all of the various services that the Scalix Plugins for Nagios can check. The primary changes that will need to be made to this file are to the "Scalix UberManager Log Scanner" and the "Scalix Mailstore /var/opt/scalix" services. The Scalix UberManager Log Scanner service includes the full path to the UberManager log files which will vary depending on where Tomcat was installed on your server and what version of Tomcat that you're using. The Scalix Mailstore service includes the filesystem name where Scalix is installed. This will vary from system to system. You can find the value on your system by looking at the results from the `df` command and looking for the filesystem where `/var/opt` is mounted. The only other changes that might be required to this file would be to change the `host_name` field (see `hosts.cfg` above) and to uncomment remote services or comment out services which are not applicable to your configuration.

```

# Service definition for the Local Scalix UberManager for Scanning Log files
define service{
    use                generic-service          ; Name of service template to use
    host_name          Scalix-Server-01
    service_description Scalix UberManager Log Scanner
    is_volatile        0
    check_period       24x7
    max_check_attempts 1
    normal_check_interval 15
    retry_check_interval 1
    contact_groups     Scalix-Admins
    notification_interval 120
    notification_period 24x7
    notification_options w,u,c,r
    check_command      check_log!/var/opt/jakarta-tomcat-
5.0.27/logs/caa.log!/tmp/caa.log.old!FATAL
}

#Service definition for Check Scalix Message Store FS
define service{
    use                generic-service          ; Name of service template to use
    host_name          Scalix-Server-01
    service_description Scalix Mailstore /var/opt/scalix
    is_volatile        0
    check_period       24x7
    max_check_attempts 3
    normal_check_interval 5
    retry_check_interval 1
    contact_groups     Scalix-Admins
    notification_interval 120
    notification_period 24x7
    notification_options w,u,c,r
    check_command      check_local_disk!20%!10%!/dev/vg1/LogVol100
}

```

### 5.2.4. Contact Definitions

The `contacts.cfg` file defines the people who will be notified if Nagios detects a problem. The only fields that need to be modified in this file are the email and pager fields.

```

# Scalix contact definition
define contact{

```

```

contact_name      scalix-admin
alias             Scalix Admin
service_notification_period 24x7
host_notification_period 24x7
service_notification_options w,u,c,r
host_notification_options d,u,r
service_notification_commands notify-by-email,notify-by-epager
host_notification_commands host-notify-by-email,host-notify-by-epager
email            scalix-admin@localhost.localdomain
pager            pagescalix-admin@localhost.localdomain
}

```

### 5.2.5. Host Group Definitions

The `hostgroup.cfg` file defines a group of hosts that can be grouped together to simplify notification. When a problem occurs with a Scalix server, Nagios will find which Host Group the problematic server belongs to, search the Contact Groups (see below) for this Host Group, then notify all Contacts associated with that Contact Group.

If you chose to use the default name for the Contact Group, you will only need to change the list of members. The default list of members includes the three Scalix servers defined in the `hosts.cfg` file.

```

# Scalix Server host group definition
define hostgroup{
    hostgroup_name Scalix-Servers
    alias          Scalix Servers
    contact_groups Scalix-Admins
    members        Scalix-Server-01,Scalix-Server-02,Scalix-Server-03
}

```

### 5.2.6. Contact Group Definitions

The `contactgroup.cfg` file defines a group of contacts that can be grouped together to simplify notification. When a problem occurs with a Scalix server, Nagios will find which Host Group (see above) the problematic server belongs to, search the Contact Groups for this Host Group, then notify all Contacts associated with that Contact Group.

If you chose to use the default Contact Name (see above), you will not need to make any changes to this file.

```

# Scalix Admins contact group definition
define contactgroup{
    contactgroup_name Scalix-Admins
    alias             Scalix Administrators
    members           Scalix-Admin
}

```

### 5.2.7. Command Definitions

The `checkcommands.cfg` file defines the actual commands used in the `services.cfg` file (see above). The only changes that may be required are the port numbers where UberManager, RES and SWA are listening.

```

# 'check_ubermanager' command definition
define command {
    command_name check_ubermanager
    command_line $USER1$/check_http -H $HOSTADDRESS$ -u /caa?query=monitor -p 8080
}
# 'check_res' command definition
define command {
    command_name check_res
    command_line $USER1$/check_http -H $HOSTADDRESS$ -u /res/RESDispatcher?query=monitor -p
8080
}
# 'check_swa' command definition
define command {

```

```

command_name    check_swa
command_line    $USER1$/check_http -H $HOSTADDRESS$ -u /webmail/ -p 8080
}

```

### 5.2.8. Escalation Definitions

The escalations.cfg file is used to escalate notifications for all hosts in the Scalix Host Groups (see above). If you use the default names for the Scalix Host and Contact Group, no changes are required to this file.

```

define hostgroupescalation{
    hostgroup_name    Scalix-Servers
    first_notification    2
    last_notification    5
    contact_groups    Scalix-Admins
    notification_interval    10
}

```

## 5.3. Scalix Plugins for Nagios – Remote (nrpe)

### 5.3.1. nrpe.cfg

The nrpe.cfg file is similar to the Command Definitions file above, but designed for remote monitoring. The Scalix specific entries in this are listed below. The two lines which will likely need to be customized are the check\_mailstore and check\_log commands. These changes are the same as for the Service Definitions above.

```

command[check_swap]=/usr/lib/nagios/plugins/check_swap -w 85% -c 95%
command[check_mailstore]=/usr/lib/nagios/plugins/check_disk -w 20 -c 10 -p /dev/vgscalix/lvscalix
command[check_local_sxqueue]=sudo /usr/lib/nagios/plugins/check_queues.py -c 100 -w 75 -q LOCAL
command[check_router_sxqueue]=sudo /usr/lib/nagios/plugins/check_queues.py -c 100 -w 75 -q ROUTER
command[check_smtp_sxqueue]=sudo /usr/lib/nagios/plugins/check_queues.py -c 100 -w 75 -q SMINTFC
command[check_error_sxqueue]=sudo /usr/lib/nagios/plugins/check_queues.py -c 100 -w 75 -q ERROR
command[check_unix_sxqueue]=sudo /usr/lib/nagios/plugins/check_queues.py -c 100 -w 75 -q UNIX
command[check_sxdaemons]=sudo /usr/lib/nagios/plugins/check_daemons.py -d
command[check_sxservices]=sudo /usr/lib/nagios/plugins/check_daemons.py -s
command[check_inodes]=/usr/lib/nagios/plugins/check_inode.py -c 80 -w 70 -f /var/opt/scalix
command[check_memory]=/usr/lib/nagios/plugins/check_memory.py -c 90% -w 80%
command[check_log]=/usr/lib/nagios/plugins/check_log -F /var/opt/jakarta-tomcat-5.0.27/logs/res.log
-O /tmp/res.log.old -q FATAL
command[check_tcp_connections]=/usr/lib/nagios/plugins/check_inet_connections.py -c 250 -w 200 -T
tcp

```