IBM® Tivoli® Netcool/OMNIbus Probe for Alcatel-Lucent 9353 WNMS/OAM (CORBA) 2.0

Reference Guide
March 12, 2015

SC27-2426-06

Notice

Before using this information and the product it supports, read the information in Appendix A, “Notices and Trademarks,” on page 21.

Edition notice

This edition (SC27-2426-06) applies to version 2.0 of IBM Tivoli Netcool/OMNIbus Probe for Alcatel-Lucent 9353 WNMS/OAM (CORBA) and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces SC27-2426-05.

Contents

About this guide

The following sections contain important information about using this guide.

Document control page

Use this information to track changes between versions of this guide.

The IBM Tivoli Netcool/OMNIbus Probe for Alcatel-Lucent 9353 WNMS/OAM (CORBA) documentation is provided in softcopy format only. To obtain the most recent version, visit IBM Documentation:

https://www.ibm.com/support/knowledgecenter/SSSHTQ_int/omnibus/common/kc_welcome-444.html

Table 1. Document modification history

Document version	Publication date	Comments
SC27-2426-00	December 31, 2008	First IBM® publication.
SC27-2426-01	February 19, 2010	Version of WNMS supported by the probe updated in "Summary" on page 1.
SC27-2426-02	May 31, 2010	"Summary" on page 1 updated.
SC27-2426-03	March 31, 2011	Installation section replaced by "Installing probes" on page 2. "Firewall considerations" on page 3 added.
SC27-2426-04	November 04, 2011	Information about operating system conventions added in "Conventions used in this guide" on page vi. Requirements and multicultural support information updated in "Summary" on page 1. CORBA connection information updated in "Connecting to the CORBA interface" on page 4. The following new properties were added in "Properties and command line options" on page 8: • **ORBCharEncoding** • **ORBWCharDefault** Information about running the probe added in "Running the probe" on page 20.
SC27-2426-05	May 03, 2013	Support extended to Alcatel-Lucent 9353 WNMS version 8.1.
SC27-2426-06	March 12, 2015	Support extended to Alcatel-Lucent 9353 WNMS version 9.1.

Conventions used in this guide

All probe guides use standard conventions for operating system-dependent environment variables and directory paths.

Operating system-dependent variables and paths

All probe guides use standard conventions for specifying environment variables and describing directory paths, depending on what operating systems the probe is supported on.

For probes supported on UNIX and Linux operating systems, probe guides use the standard UNIX conventions such as **$***variable* for environment variables and forward slashes (**/**) in directory paths. For example:

```
$OMNIHOME/probes
```

For probes supported only on Windows operating systems, probe guides use the standard Windows conventions such as **%***variable***%** for environment variables and backward slashes (****) in directory paths. For example:

```
%OMNIHOME%\probes
```

For probes supported on UNIX, Linux, and Windows operating systems, probe guides use the standard UNIX conventions for specifying environment variables and describing directory paths. When using the Windows command line with these probes, replace the UNIX conventions used in the guide with Windows conventions. If you are using the bash shell on a Windows system, you can use the UNIX conventions.

Note: The names of environment variables are not always the same in Windows and UNIX environments. For example, %TEMP% in Windows environments is equivalent to $TMPDIR in UNIX and Linux environments. Where such variables are described in the guide, both the UNIX and Windows conventions will be used.

Operating system-specific directory names

Where Tivoli Netcool/OMNIbus files are identified as located within an *arch* directory under NCHOME or OMNIHOME, *arch* is a variable that represents your operating system directory. For example:

```
$OMNIHOME/probes/arch
```

The following table lists the directory names used for each operating system.

Note: This probe may not support all of the operating systems specified in the table.

Table 2. Directory names for the arch variable

Operating system	**Directory name represented by *arch***
AIX® systems	`aix5`
Red Hat Linux® and SUSE systems	`linux2x86`
Linux for System z	`linux2s390`
Solaris systems	`solaris2`
Windows systems	`win32`

OMNIHOME location

Probes and older versions of Tivoli Netcool/OMNIbus use the OMNIHOME environment variable in many configuration files. Set the value of OMNIHOME as follows:

- On UNIX and Linux, set $OMNIHOME to $NCHOME/omnibus.
- On Windows, set %OMNIHOME% to %NCHOME%\omnibus.

Chapter 1. Probe for Alcatel-Lucent 9353 WNMS/OAM (CORBA)

The Alcatel-Lucent 9353 WNMS wireless management system offers standardized 3rd Generation Partnership Project (3GPP) communication interfaces.

The Probe for Alcatel-Lucent 9353 WNMS/OAM (CORBA) collects data from Alcatel-Lucent 9353 WNMS using a Common Object Request Broker Architecture (CORBA) 3GPP interface.

The probe complies with the following 3GPP standards:

Table 3. Supported 3GPP standards

3GPP Technical Specification	Version	Object
32.111-3	6.3.0	Alarm Integration Reference Point (IRP)
32.303	6.4.0	Notification IRP
32.363	6.3.0	Entry Point IRP

This guide contains the following sections:

Summary

Each probe works in a different way to acquire event data from its source, and therefore has specific features, default values, and changeable properties. Use this summary information to learn about this probe.

The following table provides a summary of the Probe for Alcatel-Lucent 9353 WNMS/OAM (CORBA).

Table 4. Summary

Probe target	Alcatel-Lucent 9353 WNMS versions 6.0, 7.0, 8.1, and 9.1.
Probe executable name	`nco_p_alcatel_wnms`
Probe installation package	`omnibus-`*`arch`*`-probe-nco-p-alcatel-wnms-`*`version`*
Package version	2.0

Table 4. Summary (continued)	
Probe supported on	For details of supported operating systems, see the following Release Notice on the IBM Software Support website: https://www-304.ibm.com/support/docview.wss?uid=swg21412226
Properties file	`$OMNIHOME/probes/arch/alcatel_wnms.props`
Rules file	`$OMNIHOME/probes/arch/alcatel_wnms.rules`
Requirements	For details of any additional software that this probe requires, refer to the `description.txt` file that is supplied in its download package.
Connection method	CORBA
Remote connectivity	The probe can connect to a remote device using a CORBA interface.
Multicultural support	Not Available
Peer-to-peer failover functionality	Available
IP environment	IPv4 and IPv6
Federal Information Processing Standards (FIPS)	IBM Tivoli Netcool/OMNIbus uses the FIPS 140-2 approved cryptographic provider: IBM Crypto for C (ICC) certificate 384 for cryptography. This certificate is listed on the NIST website at http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/1401val2004.htm. For details about configuring Netcool/OMNIbus for FIPS 140-2 mode, see the *IBM Tivoli Netcool/OMNIbus Installation and Deployment Guide*.

Installing probes

All probes are installed in a similar way. The process involves downloading the appropriate installation package for your operating system, installing the appropriate files for the version of Netcool/OMNIbus that you are running, and configuring the probe to suit your environment.

The installation process consists of the following steps:

1. Downloading the installation package for the probe from the Passport Advantage Online website.

   Each probe has a single installation package for each operating system supported. For details about how to locate and download the installation package for your operating system, visit IBM Documentation:

   https://www.ibm.com/support/knowledgecenter/SSSHTQ_int/omnibus/probes/all_probes/wip/reference/install_download_intro.html

2. Installing the probe using the installation package.

   The installation package contains the appropriate files for all supported versions of Netcool/OMNIbus. For details about how to install the probe to run with your version of Netcool/OMNIbus, visit the following page in IBM Documentation:

   https://www.ibm.com/support/knowledgecenter/SSSHTQ_int/omnibus/probes/all_probes/wip/reference/install_install_intro.html

3. Configuring the probe.

   This guide contains details of the essential configuration required to run this probe. It combines topics that are common to all probes and topics that are peculiar to this probe. For details about additional configuration that is common to all probes, see the *IBM Tivoli Netcool/OMNIbus Probe and Gateway Guide*.

Firewall considerations

When using CORBA probes in conjunction with a firewall, the firewall must be configured so that the probe can connect to the target system.

Most CORBA probes can act as both a server (listening for connections from the target system) and a client (connecting to the port on the target system to which the system writes events). If you are using the probe in conjunction with a firewall, you must add the appropriate firewall rules to enable this dual behavior.

There are three possible firewall protection scenarios, for which you must determine port numbers before adding firewall rules:

1. If the host on which the probe is running is behind a firewall, you must determine what remote host and port number the probe will connect to.
2. If the host on which the target system is running is behind a firewall, you must determine the incoming port on which the probe will listen and to which the target system will connect.
3. If each host is secured with its own firewall, you must determine the following four ports:
   a. The outgoing port (or port range) for the probe.
   b. The hostname and port of the target system.
   c. The outgoing port on which the target system sends events if the probe is running as a client.
   d. The incoming port on which the probe listens for incoming events.

Note: Most, but not all, CORBA probes listen on the port specified by the **`ORBLocalPort`** property. The default value for this property is 0, which means that an available port is selected at random. If the probe is behind a firewall, the value of the **`ORBLocalPort`** property must be specified as a fixed port number.

CORBA probes that use EventManager or NotificationManager objects may use different hosts and ports from those that use NamingService and EntryPoint objects. If the probe is configured to get object references from a NamingService or EntryPoint object, you must obtain the host and port information from the system administrator of the target system. When you have this information, you can add the appropriate firewall rules.

Configuring the lookup table

The probe is supplied with a lookup table that contains details of the various types of alarms that Alcatel-Lucent 9353 WNMS generates. You might need to update the rules file to include the path to the lookup table.

The lookup table contains details of the alarm names and probable causes generated by Alcatel-Lucent 9353 WNMS.

At installation, the `Corba_3gpp_V630.lookup` file supplied with the probe installation package is installed to the following location:

`$OMNIHOME/probes/includes/`

To reference the lookup table from the rules file, add the following line to the rules file:

```
include "../includes/Corba_3gpp_V630.lookup"
```

Note: The include command assumes that the probe is run from the standard `$OMNIHOME/probes/` directory. If you are running the probe from a different directory, replace ".." with the absolute directory path to the lookup table. Do not use the `$OMNIHOME` environment variable in this directory path.

Data acquisition

Each probe uses a different method to acquire data. Which method the probe uses depends on the target system from which it receives data.

The probe gathers alarms from Alcatel-Lucent 9353 WNMS using a CORBA 3GPP interface.

Data acquisition is described in the following topics:

Connecting to the CORBA interface

The probe acts as an Integration Reference Point (IRP) Manager and connects to Alcatel-Lucent 9353 WNMS using a CORBA 3GPP interface.

The probe uses the `SecurityIRPAgent` Interoperable Object Reference (IOR) file to connect to the CORBA interface. The `AlarmIRPOperation` and `NotificationIRPOperation` points form a part of the IRP agent.

You must specify values for the **Username** and **Password** properties. These values are required by the ALU Security Building Block to provide the object reference to the Entry Point IRP.

If you are using a local IOR file, specify its location using the **ALUSecurityIrpFile** property.

Using a remote IOR file

If the IOR file is on a remote host, use the **SecurityIrpFtpCommand** property to specify the FTP command that the probe must use to access the file. Use the following format to specify the command:

`ftp://`*`username`*`:`*`password`*`@`*`host`*`/`

Where:

- *username* is the user name of the account that provides access to the ALU Security IOR server.
- *password* is the encrypted password of the user.
- *host* is the host name of the ALU Security IOR server.

You must also specify an FTP password using the **FtpPassword** property.

Note: The FTP command is executed in the user's home directory. For FTP users other than `root`, you must change the default path specified by the **ALUSecurityIrpFile** property to enable the probe to locate the Security IRP file. For example, if the FTP user name is `user1`, the home directory for this user will be `/home/user1`.

Encrypting passwords

You must encrypt the passwords used by the **Password**, **SecurityIrpFtpCommand**, and **FtpPassword** properties, using the `nco_aes_crypt` utility supplied with Netcool/OMNIbus. Use the value returned by `nco_aes_crypt` as the value of the property in the properties file. For more information about encrypting passwords, see the *IBM Tivoli Netcool/OMNIbus Administration Guide* (SC14-7605).

Retrieving objects

If the **Resynch** property is set to true, the probe initially receives a list of all active alarms from the AlarmIRP point. The probe then connects to the NotificationIRP point and uses the CORBA notification push model to receive new alarms. If the **Resynch** property is set to false, the probe only receives new alarms.

The probe initially receives a list of all active alarms from the AlarmIRP point. The probe then connects to the NotificationIRP point and uses the CORBA notification push model to receive new alarms.

Status checking

The probe checks the status of the IRP agent every 60 seconds. You can change this frequency if required using the **Agentheartbeat** property.

Filtering notifications and alarms

The **NotificationFilter** and **AlarmFilter** properties allow you to specify what notifications and alarms are sent to the probe. When you use these properties, you must use the actual token names. For example, the token h represents the element NV_PERCEIVED_SEVERITY; so, to specify that the probe is only sent notifications with a perceived severity of 3, you must set the **NotificationFilter** property to $h == 3.

You can specify multiple filters by separating each filter with a comma; for example, to specify that the probe is sent alarms with a root of 3G, a subnet of Alcatel Lucent, and a managed element of EM RNC RNCM1_421, set the **AlarmFilter** property to:

```
$e == 'Root=3G,SubNetwork=paris,SubNetwork=Alcatel-Lucent,ManagedElement=EM RNC RNCM1_421'
```

The following table displays the token mappings for use with the **AlarmFilter** and **NotificationFilter** properties.

Table 5. Token mappings

Element	Token
NV_NOTIFICATION_ID	a
NV_EVENT_TIME	b
NV_SYSTEM_DN	c
NV_MANAGED_OBJECT_CLASS	d
NV_MANAGED_OBJECT_INSTANCE	e
NV_ALARM_ID	f
NV_PROBABLE_CAUSE	g
NV_PERCEIVED_SEVERITY	h
NV_SPECIFIC_PROBLEM	i
NV_ADDITIONAL_TEXT	j
NV_ACK_TIME	k

Table 5. Token mappings (continued)	
Element	**Token**
NV_ACK_USER_ID	l
NV_ACK_SYSTEM_ID	m
NV_ACK_STATE	n
NV_COMMENTS	o
NV_BACKED_UP_STATUS	p
NV_BACK_UP_OBJECT	q
NV_THRESHOLD_INFO	r
NV_TREND_INDICATION	s
NV_STATE_CHANGE_DEFINITION	t
NV_MONITERED_ATTRIBUTES	u
NV_PROPOSED_REPAIR_ACTIONS	v
NV_CORRELATED_NOTIFICATIONS	w
NV_REASON	x
CLEAR_USER_ID	y
CLEAR_SYSTEM_ID	z
NV_ALARM_LIST_ALIGNMENT_REQUIREMENT	ff
NV_SERVICE_USER	gg
NV_SERVICE_PROVIDER	hh
NV_SECURITY_ALARM_DETECTOR	ii
NV_VENDOR_SPECIFIC_ALARM_TYPE	jj

Command line interface

The Probe for Alcatel-Lucent 9353 WNMS/OAM (CORBA) is supplied with a command line interface (CLI). This interface allows you to execute commands using the probe (for example, to acknowledge alarms or to request a full resynchronization of the 3GPP interface). To use the CLI, you must use the **CommandPort** property in the properties file to specify a port through which commands will be sent. When you want to execute commands, telnet to this port.

The following table describes the commands that you can use with the command line interface.

Table 6. CLI commands	
Command	**Description**
`exit/quit`	This command closes the connection.
`help`	This command displays online help about the CLI.
`resynch_all`	This command allows you to perform a full resynchonization with the 3GPP interface.
`resynch_filter`	This command allows you to a perform partial resynchronization with the 3GPP interface. **Note:** This command takes as a parameter a filter in the same format as the **`AlarmFilter`** property. For details, see "Filtering notifications and alarms" on page 5.
`userid_acknowledge_alarm`	This command allows you to acknowledge an alarm in the 3GPP interface by specifying the *NV_ALARM_ID* of the alarm being acknowledged and the *NV_ACK_USER_ID*. These parameters are specified by the **`AckSystemId`** and **`AckUserId`** properties, respectively.
`userid_clear_alarm`	This command allows you to clear the alarm by the user specified.
`userid_unacknowledge_alarm`	This command allows you to unacknowledge an alarm in the 3GPP interface by specifying the *NV_ALARM_ID* of the alarm being acknowledged and the *NV_ACK_USER_ID*.
`version`	This command displays the version of the probe.

Peer-to-peer failover functionality

The probe supports failover configurations where two probes run simultaneously. One probe acts as the `master` probe, sending events to the ObjectServer; the other acts as the `slave` probe on standby. If the master probe fails, the slave probe activates.

While the slave probe receives heartbeats from the master probe, it does not forward events to the ObjectServer. If the master probe shuts down, the slave probe stops receiving heartbeats from the master and any events it receives thereafter are forwarded to the ObjectServer on behalf of the master probe. When the master probe is running again, the slave probe continues to receive events, but no longer sends them to the ObjectServer.

Example property file settings for peer-to-peer failover

You set the peer-to-peer failover mode in the properties files of the master and slave probes. The settings differ for a master probe and slave probe.

Note: In the examples, make sure to use the full path for the property value. In other words replace `$OMNIHOME` with the full path. For example: `/opt/IBM/tivoli/netcool`.

The following example shows the peer-to-peer settings from the properties file of a master probe:

```
Server      :    "NCOMS"
RulesFile   :    "master_rules_file"
MessageLog  :    "master_log_file"
PeerHost    :    "slave_hostname"
PeerPort    :    6789 # [communication port between master and slave probe]
```

```
Mode        :    "master"
PidFile     : "master_pid_file"
```

The following example shows the peer-to-peer settings from the properties file of the corresponding slave probe:

```
Server      :    "NCOMS"
RulesFile   :    "slave_rules_file"
MessageLog  :    "slave_log_file"
PeerHost    :    "master_hostname"
PeerPort    :    6789 # [communication port between master and slave probe]
Mode        :    "slave"
PidFile     : "slave_pid_file"
```

Running multiple probes

You can run multiple instances of the probe.

For each running instance, specify a different port to which the server listens using the **ORBLocalPort** property.

Running multiple probes in a failover configuration

If you have implemented a peer-to-peer failover configuration, both the master probe and the slave probe have their own Object Request Broker (ORB) local port.

Where you are running this configuration in conjunction with a firewall, add the receiving port of each probe to the firewall rules. The direction of the connection is from the target system to the master or slave probe.

You set the peer-to-peer failover mode in the properties files of the master and slave probes.

Include the following peer-to-peer settings in the master's probe properties file:

```
PeerHost    :    "slave_hostname"
PeerPort    :    5555 # [communication port between master and slave probes]
```

Include the following peer-to-peer settings in the slave's probe properties file:

```
PeerHost    :    "master_hostname"
PeerPort    :    5555 # [communication port between master and slave probes]
```

Properties and command line options

You use properties to specify how the probe interacts with the device. You can override the default values by using the properties file or the command line options.

The following table describes the properties and command line options specific to this probe. For information about default properties and command line options, see the *IBM Tivoli Netcool/OMNIbus Probe and Gateway Guide* (SC14-7608).

Table 7. Properties and command line options

Property name	Command line option	Description
AckSystemId *string*	`-acksystemid` *string*	Use this property to specify the processing system on which the IRP manager runs. This is used by the `acknowledge_alarm` CLI function. The default is `""`.

Table 7. Properties and command line options (continued)

Property name	Command line option	Description
AckUserId *string*	`-ackuserid` *string*	Use this property to specify the name of the user acknowledging the alarm. This is used by the `acknowledge_alarm` CLI function. The default is `""`.
Agentheartbeat *integer*	`-agentheartbeat` *integer*	Use this property to specify the frequency (in seconds) with which the probe checks the status of the IRP agent. The default is `60`.
AlarmFilter *string*	`-alarmfilter` *string*	Use this property to specify the filter the alarm IRP uses to limit the alarms sent to the probe. The default is `""`.
AlarmIrpName *string*	`-alarmirpname` *string*	Use this property to specify the name of the Alarm IRP Agent. The default is `32.111-3 V6.3`.
ALUSecurityIrpFile *string*	`-alusecurityirpfile` *string*	Use this property to specify the location of the ALU Security IRP IOR file. The default is `/opt/nortel/config/3gpp/ior/ALUSecurityIrpAgent.ior`.
ClearSystemId *string*	`-clearsystemid` *string*	Use this property to specify the processing system on which the IRP manager runs. This is used by the `userid_clear_alarm` CLI function. The default is `""`.
ClearUserId *string*	`-clearuserid` *string*	Use this property to specify the name of the user acknowledging the alarm. This is used by the `user_clear_alarm` CLI function. The default is `""`.
CommandPort *integer*	`-commandpor` *integer*	Use this property to specify the port to which users can telnet to communicate with the 3GPP interface using the CLI supplied with the probe. For details about the CLI, see “Command line interface” on page 6. The default is `6970`.

Table 7. Properties and command line options (continued)

Property name	Command line option	Description
CommandPortLimit *integer*	`-commandportlimit` *integer*	Use this property to specify the maximum number of telnet connections that can be made to the probe. The default is `10`.
EntryPointIrpName *string*	`-entrypointirpname` *string*	Use this property to specify the name used to resolve the entry point IRP in the Naming Service. The default is `32.363 V6.3`.
FlushBufferInterval *integer*	`-flushbufferinterval` *integer*	Use this property to specify how often (in seconds) the probe flushes all alerts in the buffer to the ObjectServer. The default is `0` (which instructs the probe to never flush the alerts to the ObjectServer).
FtpPassword *string*	`-ftppassword` *string*	Use this property to specify the password required to access the ALU Security IRP IOR file via FTP. The default is "". **Note:** The password must be encrypted using the `nco_aes_crypt` utility. For details about this utility, see the *IBM Tivoli Netcool/OMNIbus Administration Guide* (SC14-7605).
NotificationCategories *string*	`-notification categories` *string*	Use this property to specify the notification categories to which the probe subscribes. If the probe subscribes to more than one category, group the categories, and separate each category by a semi-colon, and place the list within quotation marks. The default is `""` (subscribes to all the notification categories).
NotificationFilter *string*	`-notificationfilter` *string*	Use this property to specify the filter the notification IRP uses to limit the notifications sent to the probe. The default is `""`.
NotificationIrpName *string*	`-notificationirpname` *string*	Use this property to specify the name of the Notification IRP Agent. The default is `32.303 V6.4`.

Table 7. Properties and command line options (continued)

Property name	Command line option	Description
ORBCharEncoding *string*	`-orbcharencoding` *string*	Use this property to specify the native character encoding set used by the Object Request Broker (ORB) for character data. The default is UTF8.
ORBLocalHostName *string*	`-orblocalhostname` *string*	Use this property to specify the local host name used by the server-side ORB to place the server's host name into the IOR of a remote object. The default is `""`.
ORBLocalPort *integer*	`-orblocalport` *integer*	Use this property to specify the local port to which the ORB listens. The default is `0` (ORB selects an available port at random).
ORBWCharDefault *string*	`-orbwchardefault` *string*	Use this property to specify what wide character (wchar) set the IBM ORB uses when communicating with other ORBs that do not publish a wchar set. The default is UTF16.
Password *string*	`-password` *string*	Use this property to specify the password required by the ALU Security Building Block to provide the object reference to the Entry Point IRP. The default is `""`. **Note:** The password must be encrypted using the `nco_aes_crypt` utility. For details about this utility, see the *IBM Tivoli Netcool/OMNIbus Administration Guide* (SC14-7605).
Resynch *string*	`-noresynch` (This is equivalent to **Resynch** with a value of `false`.) `-resynch` (This is equivalent to **Resynch** with a value of `true`.)	Use this property to specify whether the probe attempts to resynchronize alarms collected in the system at the timeout period: `false`: The probe does not resynchronize the alarms. `true`: The probe attempts to resynchronize the alarms. The default is `false`.

Table 7. Properties and command line options (continued)		
Property name	**Command line option**	**Description**
Retry *string*	`-noretry` (This is equivalent to **Retry** with a value of `false`.) `-retry` (This is equivalent to **Retry** with a value of `true`.)	Use this property to specify whether the probe attempts to reconnect to the system following a timeout: `false`: The probe does not reconnect to the system. `true`: The probe attempts to reconnect to the system. The default is `false`.
SecurityIrpFtpCommand *string*	`-securityirpftpcommand` *string*	Use this property to specify the command the probe uses to get the security IRP file using FTP. The default is `""`.
Timeout *integer*	`-timeout` *integer*	Use this property to specify the time (in seconds) the probe waits to receive events before disconnecting from the Notification Service and shutting down. The default is `0` (probe never times out).
TimeTick *integer*	`-timetick` *integer*	Use this property to specify the length (in minutes) of the notification IRP session lifetime. This is used by the 3GPP server. The default is 15.
Username *string*	`-username` *string*	Use this property to specify the user name required by the ALU Security Building Block to provide the object reference to the Entry Point IRP. The default is `""`.

Elements

The probe breaks event data down into tokens and parses them into elements. Elements are used to assign values to ObjectServer fields; the field values contain the event details in a form that the ObjectServer understands.

The following table describes the elements that the Probe for Alcatel-Lucent 9353 WNMS/OAM (CORBA) generates. Not all the elements described are generated for each event. The elements that the probe generates depend on the event type.

Table 8. Elements	
Element name	**Description**
`CLEAR_SYSTEM_ID`	This element identifies whether the system identifier has been cleared.

Table 8. Elements (continued)

Element name	Description
CLEAR_USER_ID	This element identifies whether the user identifier has been cleared.
NV_ACK_STATE	This element contains the acknowledgement state of the alarm.
NV_ACK_SYSTEM_ID	This element contains the system ID of the IRP Manager processing the notification.
NV_ACK_TIME	This element contains the time at which the user acknowledged the alarm.
NV_ACK_USER_ID	This element identifies the last user who has changed the acknowledgement state.
NV_ADDITIONAL_TEXT	This element contains the information about the network element from which the alarm originated.
NV_ALARM_ID	This element contains the identification information of the alarm as it appears in the alarm list.
NV_ALARM_LIST_ALIGNMENT_REQUIREMENT	This element indicates whether the alarm list requires alignment.
NV_BACK_UP_OBJECT	This element contains the distinguished name (DN) of the backup object.
NV_BACKED_UP_STATUS	This element identifies whether the object has been backed up.
NV_COMMENTS	This element contains comments about the alarms.
NV_CORRELATED_NOTIFICATIONS	This element contains a set of notifications to which this notification is considered to be correlated. This element is generated dynamically and its content is dependent on the IRP Agent.
NV_EVENT_TIME	This element contains the time at which the event occurred.
NV_MANAGED_OBJECT_CLASS	This element contains the managed object class of the network resource.
NV_MANAGED_OBJECT_INSTANCE	This element contains the managed object instance of the network resource.
NV_MONITERED_ATTRIBUTES	This element contains the managed object attributes of the network resource.
NV_NOTIFICATION_ID	This element contains identification information of the notification.

Table 8. Elements (continued)	
Element name	**Description**
NV_PERCEIVED_SEVERITY	This element contains the relative level of urgency for operator attention.
NV_PROBABLE_CAUSE	This element provides information about the probable cause of the alarm.
NV_PROPOSED_REPAIR_ACTIONS	This element contains the proposed repair actions associated with the notification.
NV_REASON	This element contains the reason that triggered the proposed repair action.
NV_SECURITY_ALARM_DETECTOR	This element contains the security alarm detector for the device.
NV_SERVICE_PROVIDER	This element contains the name of the service provider.
NV_SERVICE_USER	This element contains the name of the service user.
NV_SPECIFIC_PROBLEMS	This element contains further information about the problem to which the notification relates.
NV_STATE_CHANGE_DEFINITION	This element contains information about the state change.
NV_SYSTEM_DN	This element contains the distinguished name (DN) used to identify the system.
NV_THRESHOLD_INFO	This element provides information about a threshold that has been crossed.
NV_TREND_INDICATION	This element indicates how an observed condition has changed.
NV_VENDOR_SPECIFIC_ALARM_TYPE	This element contains the alarm type specific to the vendor.

Error messages

Error messages provide information about problems that occur while running the probe. You can use the information that they contain to resolve such problems.

The following table describes the error messages specific to this probe. For information about generic error messages, see the *IBM Tivoli Netcool/OMNIbus Probe and Gateway Guide* (SC14-7608).

Table 9. Error messages

Error	Description	Action

`[Command Port] Failed to get property CommandPort`
`[Command Port] Error occurred`
`[Command Port] <+host +> Failed to get socket IO`
`[Command Port] <+host +> Failed to read command`
`[Command Port] host Failed to close command socket`
	There was a problem with command port functionality.	Check that you have specified the command port correctly. Check the connection between the probe and the command port.

`[Command Port] Failed to get CommandPortLimit property - using 10`	The probe could not retrieve the value of the **`CommandPortLimit`** property. The probe will use the default value of `10`.	Check that you have specified the command port correctly. Check the connection between the probe and the command port.
`[Command Port] Failed to open listening socket, shutting down Thread!`	The probe could not open a listening socket on the command port.	Try using another port.
`[Command Port] Thread shutting down due to error!`	There was a problem with command port functionality.	Check that the command port has been set, and that the `nco_p_nonnative_base` process is running.
`Cannot initialize the Orb`	Problem during initialization of the ORB.	Ensure your CLASSPATH contains the path to the Visibroker jar files.
`Cannot proceed. Shutting down!`	There is a problem with your network, or the probe configuration.	Contact IBM Software Support.

Table 9. Error messages (continued)

Error	Description	Action

`Communication failure - lost connection to NoticiationIRP:`
`CORBA.TRANSIENT exception raised. NotificationIRP is down!`
`CORBA.OBJECT_NOT_EXIST exception raised. NotificationIRP is down!`
`BAD_PARAM Exception i.e one or more of the in/out parameter is null`
`CORBA Exception stack trace to stderr`
`NetcoolIRPManager: Stack Trace to stderr`
	There is a problem with the CORBA interface.	Refer to your CORBA documentation.

`Failed to Connect`	Either the server is not running, the IOR is out of date, or the probe cannot reach the remote server.	Check that the properties are set correctly and that the target host is working correctly.
`Failed to iterate through resynch alarms`	A problem occurred while the probe was parsing the alarms retrieved during the resynch.	Check that the server is running correctly. Check that you have specified the resynchronization parameters correctly.
`Failed to narrow Security IRP interface`	The method to authenticate the caller has failed.	Check the name of the security IRP object.
`Failed to perform resynch`	The probe failed to get the alarm list, or failed to iterate through resynchronization alarms.	Check the value specified for the **Resynch** property.
`Failed to ping notification service`	The probe has connection problems with the notification IRP point.	Check the value specified for the **Agentheartbeat** property.
`Failed to resolve the AlarmIRP object`	The alarm IRP object is not registered in the Naming Service with the alarm IRP name provided in the properties file.	Check the value specified for the **AlarmIrpName** property.

Table 9. Error messages (continued)

Error	Description	Action
`Failed to resolve the NotificationIRP object`	The notification IRP object is not registered in the Naming Service with the notification IRP name provided in the properties file.	Check the value specified for the **NotificationIrpName** property.
`login: Failed to get IRP object login: Unknown exception occurred`	The probe could not get the IRP Object.	Check the name and path of the Security IRP object.
`NetcoolIRPManager: ERROR when parsing a notification event`	The probe encountered corrupted data while parsing. The data is not in expected format.	Check the settings in the notification and problem log files.
`NetcoolIRPManager: Failed to acknowledge_alarms()`	The probe could not acknowledge the alarm.	Check that the alarm identifier provided to the CLI is correct.
`NetcoolIRPManager: Failed to find IOR file ior file`	The probe could not locate the IOR file.	Check the value specified for the **ALUSecurityIrpFile** property.
`NetcoolIRPManager: Failed to retrieve AlarmIRP object from security interface`	The probe could not get the alarm IRP object.	Check the value specified for the **AlarmIrpName** property.
`NetcoolIRPManager: Failed to retrieve NotificationIRP object from security interface`	The probe could not get the Notification IRP object.	Check the value specified for the **NotificationIrpName** property.
`NetcoolIRPManager: Failed to send event`	The probe could not forward the event to the ObjectServer.	Check that the ObjectServer is running.
`NetcoolIRPManager: Failed to Unacknowledge_alarms()`	The probe could not unacknowledge the alarm.	Check the connection between the probe and the command port. Check that you have specified the command port correctly.
`OperationNotSupported Exception`	The Nortel system does not support the operation specified.	Check the value specified for the **NotificationIrpName** property.
`Problem while trying to connect to the IRP points`	A problem occurred while connecting to the alarm IRP or the notification IRP.	Check the value specified for the **AlarmIrpName** and **NotificationIrpName** properties.

Table 9. Error messages (continued)		
Error	**Description**	**Action**

`Unexpected fatal error, failed to connect`
`Unexpected fatal error when connecting to interfaceUnexpected fatal error while getting IRP Outline`
`Unexpected fatal error when getting IRP reference from Entry Point IRP`
	The probe has encountered a fatal error.	Contact IBM Software Support.

ProbeWatch messages

During normal operations, the probe generates ProbeWatch messages and sends them to the ObjectServer. These messages tell the ObjectServer how the probe is running.

The following table describes the raw ProbeWatch error messages that the probe generates. For information about generic ProbeWatch messages, see the *IBM Tivoli Netcool/OMNIbus Probe and Gateway Guide* (SC14-7608).

Table 10. ProbeWatch messages		
ProbeWatch message	**Description**	**Triggers/causes**

`Communication failure: Lost connection to NotifiationIRP`
`CORBA.TRANSIENT Exception: Lost connection to NotificationIRP`
`CORBA.OBJECT_NOT_EXIST Exception: Lostconnection to NotificationIRP`
`Failed to connect to NotificationIRP: Shutting down`
`Failed to Connect: Either, the server is not running, the IOR is out of date, or probe cannot reach remote Server`
	The communication with the notification IRP server has failed.	The probe lost its connection to the notification IRP server.

`END SYNCHRONIZATION`	The synchronization of alarms has stopped.	The probe has resynched all the available alarms in the server.

Table 10. ProbeWatch messages (continued)

ProbeWatch message	Description	Triggers/causes
`Failed to find IOR file alarmirp_ior_file`	The specified alarm IRP file does not exist.	The IRP manager has been set up incorrectly.
`Failed to find IOR file notificationirp_ior_file`	The specified notification IRP file does not exist.	The IRP manager has been set up incorrectly.
`Failed to get IRP Outline (GetIRPOutline exception caught when calling get_IRP_outline)`	The probe could not get the IRP object.	There is a problem with the CORBA interface.
`Failed to get IRP reference (GetIRPReference exception caught when calling get_IRP_reference` `Failed to get IRP references - invalid parameter (InvalidParameter exception caught when calling get_IRP_reference).`	The probe could not get the IRP version.	An incorrect reference is specified in the IRP file.
`Failed to listen for commands on port number listening_port`	A problem occurred while trying to listen for commands.	Either there was a problem initializing the connection due to insufficient memory or (if this message was sent after some events had been parsed) there was a license or a connection failure.
`Failed to log in to interface (InvalidParameter exception caught when calling get_IRP_outline)`	The probe failed to login to the server.	The specified user name and password are incorrect.
`Failed to log in to the Nortel SecurityIRP - incorrect user name/ password`	The probe has failed to login to the Nortel Security IRP.	The password entered is mismatching the user name.
`Invalid IRPVersion (InvalidIRPVersion exception caught when calling get_IRP_outline)`	The specified IRP version is incorrect.	An incorrect IRP name is specified in the properties file.
`NetcoolIRPManager: Failed to find IOR file ior_file`	The specified IOR file does not exist.	The IRP Manager has been set up incorrectly.

Table 10. ProbeWatch messages (continued)		
ProbeWatch message	**Description**	**Triggers/causes**
`PermissionDenied (PermissionDenied exception caught when calling get_IRP_outline)`	The permission to login to the server is denied.	The probe does not have appropriate permissions set to log in to the server.
`START SYNCHRONIZATION`	The synchronization of alarms has started.	The probe started to resynchronize alarms collected in the system during the timeout period.
`Will listen for commands on port number listening_port`	The probe will listen for commands on the specified port number.	The probe has successfully created the specified command port in the properties file.

Running the probe

Before running the probe for the first time, you must specify a minimum set of properties.

You must specify values for the following properties before running the probe:

- **ALUSecurityIrpFile**
- **FtpPassword** (if you are obtaining the ALU Security IOR file by FTP)
- **Password**
- **SecurityIrpFtpCommand** (if you are obtaining the ALU Security IOR file by FTP)
- **Username**

For more information about these properties, see "Connecting to the CORBA interface" on page 4.

Starting the probe

To start the probe, use the following command:

```
$OMNIHOME/probes/nco_p_alcatel_wnms
```

Shutting down the probe

To stop an instance of the probe, issue a stop signal to the process associated with that probe instance.

Appendix A. Notices and Trademarks

This appendix contains the following sections:

- Notices
- Trademarks

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing 2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who want to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Software Interoperability Coordinator, Department 49XA

3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, ibm.com, AIX, Tivoli®, zSeries, and Netcool are trademarks of International Business Machines Corporation in the United States, other countries, or both.

Adobe, Acrobat, Portable Document Format (PDF), PostScript, and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Intel, Intel Inside (logos), MMX, and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java™ and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

SC27-2426-06