

IBM® Tivoli® Netcool/OMNIbus Probe for  
Lucent 5ESS  
5.0

*Reference Guide*  
*March 31, 2011*



**Note**

Before using this information and the product it supports, read the information in [Appendix A, “Notices and Trademarks,”](#) on page 15.

**Edition notice**

This edition applies to version 5.0.2 of IBM Tivoli Netcool/OMNIbus Probe for Alcatel-Lucent 5ESS (SC23-7874-03) and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces SC23-7874-02.

© **Copyright International Business Machines Corporation 2006, 2011.**

US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

---

# Contents

- Document control page..... V**
  
- Chapter 1. Probe for Alcatel-Lucent 5ESS..... 1**
  - Summary..... 1
  - Installing probes..... 2
  - Internationalization support..... 2
    - Example multi-byte character set on Solaris..... 3
  - Data acquisition..... 4
    - Chat in and chat out scripts..... 4
    - Backoff strategy..... 4
    - ReadTimeout..... 5
    - Inactivity..... 5
    - Invalid alarm logging..... 5
    - Data stream capture..... 5
    - Peer-to-peer failover functionality..... 6
  - Properties and command line options..... 6
    - Escape codes..... 11
  - Elements..... 11
  - Error messages..... 13
  - ProbeWatch messages..... 13
  
- Appendix A. Notices and Trademarks..... 15**
  - Notices..... 15
  - Trademarks..... 16



## Document control page

---

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

The IBM Tivoli Netcool/OMNIBus Probe for Alcatel-Lucent 5ESS documentation is provided in softcopy format only. To obtain the most recent version, visit the IBM® Tivoli® Information Center:

[http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/index.jsp?topic=/com.ibm.tivoli.nam.doc/welcome\\_ptsm.htm](http://publib.boulder.ibm.com/infocenter/tivihelp/v8r1/index.jsp?topic=/com.ibm.tivoli.nam.doc/welcome_ptsm.htm)

<b>Document version</b>	<b>Publication date</b>	<b>Comments</b>
01	December 31, 2008	Summary table updated. FIPS information added. Installation section added.
02	June 04, 2010	Summary table updated. Installation section updated.
03	March 31, 2011	Installation section replaced by <a href="#">“Installing probes”</a> on page 2.



# Chapter 1. Probe for Alcatel-Lucent 5ESS

Alcatel-Lucent 5ESS is a multiservice switching system. It forms part of the Lucent 7 R/E Networks architecture that provides packet and voice network functionality.

This guide contains the following sections:

- “[Summary](#)” on page 1
- “[Installing probes](#)” on page 2
- “[Internationalization support](#)” on page 2
- “[Data acquisition](#)” on page 4
- “[Properties and command line options](#)” on page 6
- “[Elements](#)” on page 11
- “[Error messages](#)” on page 13
- “[ProbeWatch messages](#)” on page 13

## 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 5ESS.

Probe target	Lucent 5ESS
Probe executable name	nco_p_5ess
Package version	5.0
Probe supported on	For details of the operating systems on which the probe is supported, see the following Release Notice on the IBM Software Support website: <a href="https://www-304.ibm.com/support/docview.wss?uid=swg21432736">https://www-304.ibm.com/support/docview.wss?uid=swg21432736</a>
Properties file	\$OMNIBUS_HOME/probes/arch/5ess.props
Rules file	\$OMNIBUS_HOME/probes/arch/5ess.rules
Requirements	A currently supported version of IBM Tivoli Netcool/OMNIBus.
Connection method	TCP/IP
Remote connectivity	The Probe for Alcatel-Lucent 5ESS can connect to a device on a remote host. Details of the remote device are specified using the <b>Host</b> and <b>Port</b> properties in the properties file.
Multicultural support	Not Available

<i>Table 2. Summary (continued)</i>	
Peer-to-peer failover functionality	Available
IP environment	IPv4 only
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 <a href="http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/1401val2004.htm">http://csrc.nist.gov/groups/STM/cmvp/documents/140-1/1401val2004.htm</a> . For details about configuring Netcool/OMNIBus for FIPS 140-2 mode, see the <i>IBM Tivoli Netcool/OMNIBus Installation and Deployment Guide</i> .

## 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 the following page on the IBM Tivoli Knowledge Center:

[http://www-01.ibm.com/support/knowledgecenter/SSSHTQ/omnibus/probes/all\\_probes/wip/reference/install\\_download\\_intro.html](http://www-01.ibm.com/support/knowledgecenter/SSSHTQ/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 on the IBM Tivoli Knowledge Center:

[http://www-01.ibm.com/support/knowledgecenter/SSSHTQ/omnibus/probes/all\\_probes/wip/reference/install\\_install\\_intro.html](http://www-01.ibm.com/support/knowledgecenter/SSSHTQ/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*.

## Internationalization support

The probe supports multibyte character sets (for example, Japanese) and character sets that contain individual multibyte characters (for example German, French, and Spanish). To view the character sets correctly, you must configure the locale settings on the host machine correctly.

If you are using a language that contains multibyte characters, you must set the LANG environment variables to the name of your character set, and export the LC\_ALL environment variable. For example, if you are using Japanese, set these environment variables to ja\_JP.UTF-8; if you are using German, set these environment variables to de\_DE.UTF-8. This will enable the probe to recognise the multibyte characters used by your character set when they occur in any network events.

The probe supports the following language locales:



Table 3. Supported language locales

Languages	AIX	HP-UX	Solaris	Linux
English (US)	en_US	en_US	en_US	en_US
Simplified Chinese	zh_CN	zh_CN	zh_CN	zh_CN
Traditional Chinese	zh_TW	zh_TW.eucTW	Zh_TW.big5	zh_TW.big5
Czech	cs_CZ	cs_CZ	cs	cs_CZ
French (standard)	fr_FR	fr_FR	fr	fr_FR
German (standard)	de_DE	de_DE	de	de_DE
Hungarian	hu_HU	hu_HU	hu	hu_HU
Italian (standard)	it_IT	it_IT	it	it_IT
Japanese	ja_JP	ja_JP	ja	ja_JP
Korean	ko_KR	ko_KR	ko	ko_KR
Polish	pl_PL	pl_PL	pl	pl_PL
Portuguese (Brazilian)	pt_BR	pt_BR	pt	pt_BR
Russian	ru_RU	ru_RU	ru	ru_RU
Spanish	es_ES	es_ES	es	es_ES

## Example multi-byte character set on Solaris

The following steps describe how to configure Solaris to use the Japanese character set:

1. Install the necessary components for Japanese on to the host machine using the Solaris CD.
2. Set the LANG and LC\_ALL environment variables to ja\_JP PCK. This uses SJIS encoding.

**Note :** You may have to set the LANG in the host machine's default settings file and reboot it to make the changes take effect.

3. Make sure that the file \$OMNIHOME/platform/arch/locales/locales.dat has the following entry:

```
locale = ja_JP PCK, japanese, sjis
```

Where ja\_JP PCK is the vendor locale, japanese is the Sybase language, and sjis is the Sybase character set.

## 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 for Alcatel-Lucent 5ESS acquires event data from the Lucent 5ESS switch. It connects to the system using TCP/IP.

Data acquisition is described in the following topics:

- [“Chat in and chat out scripts” on page 4](#)
- [“Backoff strategy” on page 4](#)
- [“ReadTimeout” on page 5](#)
- [“Inactivity” on page 5](#)
- [“Invalid alarm logging” on page 5](#)
- [“Data stream capture” on page 5](#)
- [“Peer-to-peer failover functionality” on page 6](#)

### Chat in and chat out scripts

Chat in and chat out scripts control probe login and logout. These scripts are on a single line in the expect-send format (for chat in scripts) or send-expect format (for chat out scripts). You can specify chat in and chat out strings using the **ChatinString** and **ChatoutString** properties or the `-chatinstring` and `-chatoutstring` command line options.

The format is:

```
ChatinString : expect send expect send...
```

```
ChatoutString : send expect send expect...
```

**Note :** Each element in the chat strings is separated by white space. In order to send or expect a sequence that includes white space, surround the sequence with single quotes.

A typical chat in script might be:

```
.*login.*:.* anu\r\n .*assword.*:.* anu\r\n
```

The expect text can use any regular expression, while the send text can send any characters, including control characters using the standard UNIX/C escape sequences described in [“Escape codes” on page 11](#).

### Backoff strategy

If the **Retry** property is set to `true`, and the probe fails to establish a connection or loses an existing connection to the device, the probe reverts to a backoff strategy. The probe tries to reestablish a connection after one second, two seconds, then four seconds, eight seconds, and so on, up to a maximum of 4096 seconds.

After the connection is made to the specified port, the probe tries to log in to the device. If the probe fails to log in, it shuts down and tries to connect again. The backoff strategy remains in place until a successful login occurs. The user can also specify a reconnection interval using the **ReconnectionInterval** property or `-reconnectioninterval` command line option. When this property is enabled, the probe reconnects at the specified time interval instead of using the backoff strategy.

If the remote host terminates the connection, the probe closes the connection on the host machine. The operating system is not allowed to close the connection.

## ReadTimeout

The **ReadTimeout** property specifies how long the probe waits to read alarm data before timing out. Each time the probe attempts to read an alarm, this is the allotted time that it waits to receive data. If nothing is received, the probe moves on to the next alarm.

## Inactivity

The probe has a timeout facility that allows it to disconnect from the socket if it fails to receive the next alarm data within a predefined amount of time. To specify how long the probe waits before disconnecting, use the **Inactivity** property. After this length of time, the probe disconnects from the switch, sends a ProbeWatch message to the ObjectServer, and tries to reconnect.

You can also use the **InactivityRetry** property to specify the number of consecutive inactivity intervals that the probe allows before disconnecting. If this property is set to 0, the probe disconnects after a single period of inactivity.

## Invalid alarm logging

You can set the probe to write the details of any received invalid alarms to an invalid log file. To do this, set the **InvalidLogging** property to 1. To specify the name of the file, use the **InvalidAlarmLogFile** property. To specify the maximum size of the log file, use the **MaxInvalidAlarmLog** property.

If the flushtime is activated while the probe is reading an alarm, it discards the buffer. Any incomplete alarms that result following a flushtime activation are logged as invalid. To specify the length of the flushtime period, use the **FlushTime** property.

## Data stream capture

The probe can capture the data stream sent from a device in a stream capture file. For each connection, the full data stream is stored in a stream capture file

The data stream is stored using the following naming convention:

*streamcapturefile\_host\_port*

where:

- *streamcapturefile* is the value specified by the **StreamCaptureFile** property
- *host* is the name of the host to which the probe is connected
- *port* is the port on which the probe is listening for data

Stream capture data can be used for debugging purposes, to develop new features for the probe, or to pass to other management systems that require the same data.

**Note :** If you leave the **StreamCaptureFile** property blank, the data stream capture function is disabled.

## Rotating stream capture files

The probe can rotate stream capture files; that is, it can write to a stream capture file that is saved and archived periodically when a predefined file size is reached

To use this feature, set the maximum size for the stream capture file using the **MaxStreamCapFileSize** property and set the **DateStreamCapture** property to 1.

When the **DateStreamCapture** property is set to 1, the probe creates a stream capture file with the following naming convention:

*streamcapturefile\_host\_port\_date\_time*

By appending the filename with a timestamp, the probe avoids overwriting the old stream capture file.

The maximum file size specified by the **MaxStreamCapFileSize** property acts as an upper limit for the stream capture file. If the probe reads a stream whose size exceeds the remaining allowable space in the current stream capture file, it saves the current file and creates a new one, thus storing the whole stream in a single file.

If you do not specify a maximum size for the stream capture file, it grows indefinitely until the connection is closed. If the **MaxStreamCapFileSize** property is set and the **DateStreamCapture** property is set to 0, the probe overwrites the stream capture file for that connection each time the maximum file size is reached.

## 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"
```

## 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*, (SC14-7530).

Table 4. Properties and command line options

Property name	Command line option	Description
<b>ChatinString</b> <i>string</i>	-chatinstring <i>string</i>	Use this property to specify the chat in script for connection to the host system. The default is "".
<b>ChatoutString</b> <i>string</i>	-chatoutstring <i>string</i>	Use this property to specify the chat out script for disconnection from the host system. The default is "".
<b>DateStreamCapture</b> <i>integer</i>	-datestreamcapture <i>integer</i>	Use this property to specify whether or not the probe appends a date and time to the stream capture filename. This property takes the following values:  0: The probe does not append the date and time to the stream capture file name.  1: The probe appends the date and time to the stream capture file name.  The default is 0.  <b>Note :</b> Setting the <b>DateStreamCapture</b> property to 1 prevents the stream capture file from being overwritten.
<b>ExpectTimeout</b> <i>integer</i>	-expecttimeout <i>integer</i>	Use this property to specify the time (in seconds) that the probe waits for a response to the expect part of the chat in or chat out script. The default is 20.
<b>FlushTime</b> <i>integer</i>	-flushtime <i>integer</i>	Use this property to specify the time (in seconds) that the probe waits to receive an event terminator before sending an event to the ObjectServer. The default is 5.  <b>Note :</b> If this property is set to 0, the probe only sends events to the ObjectServer after it has received their event terminator.

Table 4. Properties and command line options (continued)

Property name	Command line option	Description
<b>Host</b> <i>string</i>	-host <i>string</i>	Use this property to specify the name of the host to which the probe connects.  The default is localhost.
<b>Inactivity</b> <i>integer</i>	-inactivity <i>integer</i>	Use this property to specify the time (in seconds) that the probe allows a port to be inactive before disconnecting.  The default is 0 (the probe does not disconnect if the port becomes inactive).
<b>InactivityRetry</b> <i>integer</i>	-inactivityretry <i>integer</i>	Use this property to specify the number of consecutive periods of inactivity that the probe allows before attempting to reconnect to the host.  The default is 0 (this disables the feature).
<b>InvalidAlarmLogFile</b> <i>string</i>	-invalidalarmlogfile <i>string</i>	Use this property to specify the file to which the probe logs invalid alarms. The current date and time is appended to this filename.  The default is \$OMNIHOME/var/5ess.invalid.
<b>InvalidAlarmLogging</b> <i>integer</i>	-invalidalarmlogging <i>integer</i>	Use this property to specify whether or not the probe logs invalid alarms. This property takes the following values:  0: The probe does not log invalid alarms. 1: The probe logs invalid alarms.  The default is 0.
<b>MaxEvents</b> <i>integer</i>	-maxevents <i>integer</i>	Use this property to specify the maximum number of events that the probe attempts to simultaneously read from a given source.  The default is 10.  <b>Note :</b> You can increase this value to increase the event throughput when a large number of events is generated.

Table 4. Properties and command line options (continued)

Property name	Command line option	Description
<b>MaxInvalidAlarmLogFileSize</b> <i>integer</i>	-maxinvalidlog filesize <i>integer</i>	Use this property to specify the maximum size (in bytes) for the invalid alarm log file. When this limit is reached, the probe creates a new file with a new date and time.  The default is 1048576.
<b>MaxStreamCapFileSize</b> <i>integer</i>	- maxstreamcapfilesize <i>integer</i>	Use this property to specify the maximum size (in bytes) of the stream capture file. When this limit is reached, the probe creates a new file.  The default is 0.  <b>Note :</b> If the probe reads a stream whose size exceeds the remaining allowable space in the file, it creates a new file to store the whole stream in.
<b>Port</b> <i>integer</i>	-port <i>integer</i>	Use this property to specify the port to which the probe connects.  The default is 23.
<b>ReadTimeout</b> <i>integer</i>	-readtimeout <i>integer</i>	Use this property to specify the time (in milliseconds) that the probe waits before a socket-read times out.  The default is 100 (equal to 0.1 seconds).  <b>Note :</b> The default value for this property is set low to maximize performance. This ensures that the probe quickly moves on to the next socket. Only increase this value if you have a slow network.
<b>ReconnectionAttempts</b> <i>integer</i>	- reconnectionattempts <i>integer</i>	Use this property to specify the maximum number of times that the probe attempts to reconnect to the socket.  The default is 0 (the probe makes unlimited attempts to reconnect to the socket).

Table 4. Properties and command line options (continued)

Property name	Command line option	Description
<b>ReconnectionInterval</b> <i>integer</i>	-reconnectioninterval <i>integer</i>	Use this property to specify the time (in seconds) between successive reconnection attempts.  The default is 0 (the probe uses the standard backoff strategy).
<b>StreamCaptureFile</b> <i>string</i>	-streamcapturefile <i>string</i>	Use this property to specify the file that the probe uses to store the input data stream.  The default is "".  <b>Note :</b> Leaving this property blank disables the stream capture function. When you no longer require data for debugging, you should disable the stream capture function.
<b>Switch</b> <i>integer</i>	-switch <i>integer</i>	Use this property to specify the format of the incoming event stream. This property takes the following values:  0: The event stream is in international format.  1: The event stream is in domestic format.  The default is 0.
<b>TimeFormat</b> <i>string</i>	-timeformat <i>string</i>	Use this property to specify the format used for the time stamp.  The default is %y-%m-%d %H:%M:%S
<b>WhiteSpace</b> <i>integer</i>	-whitespace (This is equivalent to <b>WhiteSpace</b> with a value of 1; omitting this command line option is equivalent to <b>WhiteSpace</b> with a value of 0.)	Use this property to specify whether or not the probe removes white space when generating \$AdditionalLine elements. This property takes the following values:  0: The probe deletes white space.  1: The probe does not delete white space.  The default is 0.



## Escape codes

You can use C-style escape codes in the **ChatinString** and **ChatoutString** properties. This allows you to easily define whether to send escape code sequences after commands.

For example, the following chatin string sends a carriage return character (\r) after the user name and password:

```
ChatinString : ".*: user\r .*: passwd\r .*:"
```

The following table explains the character sequences that are recognized.

Escape code	Character
\b	This escape code specifies the backspace character.
\f	This escape code specifies the form-feed character.
\n	This escape code specifies the new-line character.
\r	This escape code specifies the carriage return character.
\t	This escape code specifies the tab character.
\\	This escape code specifies the backslash character.
\'	This escape code specifies the single quote character.
\"	This escape code specifies the double quote character.

**Note :** Due to the way in which the above properties are parsed, the escape sequences for backslash, single quote and double quote must be double-escaped. For example, to send a backslash character (\), use \\\.

## 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 5ESS generates. Not all the elements described are generated for each event; the elements that the probe generates depends upon the event type.

Element name	Description
\$AdditionalLinexxx	These elements contain additional lines of text that describe the alarm. The alarm can contain more than one of these elements. xxx indicates a sequential number for the element.

Table 6. Elements (continued)

Element name	Description
\$GeneralHeaderLine	This element contains the general header line of the record.
\$Host	This element contains the name of the host on which Lucent 5ESS is running.
\$MessageClass	This element contains the class of the message.
\$MessageIDnn	This element contains a token from the report header. <i>nn</i> denotes the number of the report header token. This can be any integer value from 1 to 13.
\$Port	This element contains the port on the host machine to which the probe connects.
\$Priority	This element contains the priority as defined by the Lucent machine (see the 5ESS documentation for more information). This element may be blank.
\$PRMline	This element contains the full Process Recovery Message (PRM) report generated for the alarm.
\$ProcessID	This element contains the process ID of the sending process in decimal units.
\$ReportHeaderLine	This element contains the header line of the report.
\$SequenceNumber	This element contains the sequence number of the message.
\$SoftwareReleaseAndLoad	This element contains the software release load number. <b>Note :</b> The probe only generates this element when the <b>Switch</b> property is set to 1 (indicating that the alarm is in domestic format).
\$SystemName	This element contains the name of the system.
\$TranslationDate	This element contains the date when the message was translated.
\$TranslationTime	This element contains the time when the message was translated.
\$UTC	This element contains the UNIX time of the \$TranslationDate and \$TranslationTime elements.
\$UtilityID	This element contains the utility ID of the sending process in hexadecimal units. <b>Note :</b> The probe only generates this element when the <b>Switch</b> property is set to 1 (indicating that the alarm is in domestic format).
\$WholeRecord	This element contains the entire international alarm record that the probe has read.

## 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*, (SC14-7530).

Error	Description	Action
Connection session failed	The probe failed to connect to the switch.	Check the connection to the switch and that it is running correctly.
ConvertEscapes: Error allocating memory for temp buffer	The probe failed to allocate internal storage.	Make more memory available.
Invalid alarm - no alarm terminator - discarding	The probe could not parse the alarm.	Check that the alarm has a valid alarm terminator.
Not recognized format	The incoming event stream is in a format that the probe cannot recognize.	Check the connection to the switch and that it is running correctly.
Possible problem in the stream	The event stream may have become corrupted.	Check the connection to the switch and that it is running correctly.
Read error from switch	The probe read a message from the target switch, but it could not interpret the message.	Check that the switch is working correctly.
Unable to compile regexp for string	A regular expression is incorrectly formed in the rules file.	Check the rules file for the regular expression and correct the entry.

## 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*, (SC14-7530).

ProbeWatch message	Description	Triggers/causes
Attempting to Reconnect	The probe is disconnecting and reconnecting to the host because the host has become inactive.	The probe has not received any events for the period defined by the <b>Inactivity</b> property.

Table 8. ProbeWatch messages (continued)

ProbeWatch message	Description	Triggers/causes
Cannot connect to host	The probe is unable to connect to the host.	General connection failure due either to an incorrect setting of the <b>Host</b> or <b>Port</b> property, or the host refusing a connection request from the probe.
Connection attempted	The probe is trying to establish a connection to the host.	The probe is trying to establish a connection to the host.
Connection Inactive	The probe has not received any events for the period specified by the <b>InactivityAlarm</b> property.	The probe has not received any events for the period specified by the <b>Inactivity</b> property.
Connection lost ...	The probe has lost the connection to the host.	The device has dropped the connection to the probe.
Connection succeeded	The probe has successfully established a connection to the host.	The probe has successfully established a connection to the host.
Disconnection attempted	The probe is attempting to disconnect from the host.	The probe is shutting down or has lost the connection.
Disconnection succeeded	The probe has disconnected from the host.	The probe shut down or lost the connection.
Going Down ...	The probe is shutting down.	The probe is shutting down after performing the shutdown routine.
Invalid alarm - no alarm terminator - discarding	The probe has failed to parse the alarm.	The probe could not find a valid terminator in the alarm.
Login attempted	The probe is trying to log in to the host.	The probe is attempting to log in to the host.
Login failed	The probe has failed to log in to the host.	The probe has encountered a problem while logging in. Check that the <b>ChatinString</b> property is specified correctly.
Login succeeded	The probe has logged in to the host.	The probe has logged in to the host.
Logout attempted	The probe is trying to log out from the host.	The probe has received a command to shut down.

---

## 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 wish 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.

Other company, product, or service names may be trademarks or service marks of others.









SC23-7874-03

