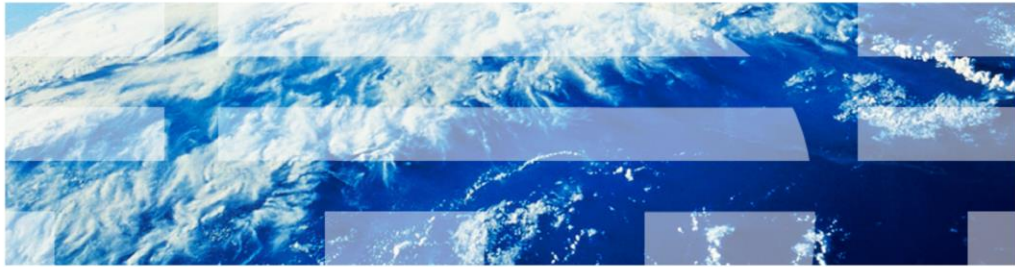


IBM Tivoli Monitoring V6.2

Reading agent logs, Part 3 - Examples, solutions, and more



© 2013 IBM Corporation

IBM Tivoli® Monitoring V6.2, Reading agent logs, part 3 examples, solutions, and more.

Assumptions

- This module is the third of three on reading Tivoli Monitoring agent logs
- This module provides examples of problems that are identified and resolved in frequently used logs. It provides suggestions on how to proceed once a suspected problem is identified
- A short description is also provided for each slide in the notes section
- Review the two previous modules before beginning this presentation:
 - *Part 1 Locating and collecting, log types, and naming conventions*
 - *Part 2 analyzing logs for error messages and keywords*
- Because a single error might have various root causes, detecting the errors might not always provide a solution, but this detection can dramatically reduce the amount of time that is required to resolve the problem

This is the third of three modules on reading Tivoli Monitoring agent logs.

In the first module you were provided with guidelines on how to collect and identify IBM Tivoli Monitoring agent logs.

The second module covers how to analyze logs and locate errors.

This module provides examples of problems identified and resolved in frequently used logs. It also provides suggestions on how to proceed once a suspected problem has been identified.

A short description is also provided for each slide in the notes section.

Review *Part 1 Locating and collecting, log types and naming conventions* and *Part 2 analyzing logs for error messages and keywords* before beginning this presentation.

This presentation works on the 80:20 principle and assumes that 80% of problems can be found in 20% of the time invested.

Because a single error might have various root causes, detecting the errors might not always provide a solution, but this detection can dramatically reduce the amount of time required to resolve the problem.

Introduction

When working with IBM Support, you can help reduce your time to resolution by being able to read log files and find problems.

The lessons covered here were developed over the past two years and this is an ongoing endeavor.

Objectives

When you complete this module, you can perform these tasks:

- Describe how error messages and keywords appear in logs
- Identify possible solutions for the errors found
- Describe how to proceed once a suspected problem is identified

When you complete this module, you can perform these tasks:

- Describe how error messages and keywords appear in logs
- Identify possible solutions for the errors found
- Describe how to proceed once a suspected problem is identified

Keyword examples, components, and solutions

This section provides real world examples of problems that were resolved with the help of keywords.

The examples shown here are from actual Problem Management Records (PMR).

Each example provides the following information:

- A customer description of the original problem
- The component involved
- The log name or names affected
- An example of the log text with the problem and keywords in bold font
- The solution, if one was determined

Example: Connected to the Tivoli Enterprise Monitoring Server or not

- Agents must connect to their assigned Tivoli Enterprise Monitoring Server soon after they are started
- If an agent successfully connects to its Tivoli Enterprise Monitoring Server, previously known as a Candle Management Server (CMS), you should see this message:
(4CBF02C2.0025-1284:kraarreg.cpp,324,"ConnectToProxy") **Successfully connected to CMS HUB_<hubname>** using ip.pipe:#<IP Address>[1918]
- If the agent fails to connect to the Tivoli Enterprise Monitoring Server, you see this error:
Error "Unable to find running CMS on CT_CMSLIST"
(2011/02/18,11:37:36.00C0-10:kraarreg.cpp,1886,"FindProxyUsingLocalLookup")
Unable to find running CMS on CT_CMSLIST <ip.pipe:xxxxxxx>

Agents must connect to their assigned Tivoli Enterprise Monitoring Server soon after they are started.

Checking for agent to Tivoli Enterprise Monitoring Server connectivity is a good place to start when analyzing logs.

These messages are near the beginning of the log.

If an agent successfully connects to its Tivoli Enterprise Monitoring Server (TEMS), previously known as a Candle Management Server (CMS), you should see a message that states "Successfully connected to CMS."

If the agent fails to connect to the Tivoli Enterprise Monitoring Server, you should see an error that indicates the agent was "Unable to find running CMS on CT_CMSLIST."

This error might have various causes but determining if the agent connects to the Tivoli Enterprise Monitoring Server helps to determine the next steps to take.

Example: Tivoli Enterprise Monitoring Server communication

- Problem Description: Universal Agent errors connecting to Tivoli Enterprise Monitoring Server
- Component: Universal agent (UM)
- Log name: <hostname>_um_<timestamp>-0#.log
- Example:


```
(4CBE1EB2.0425-FE8:kdcl0cl.c,142,"KDCL0_ClientLookup") status=1c020006, "location server unavailable",
ncs/KDC1_STC_SERVER_UNAVAILABLE
(4CBE1EB2.0426-FE8:kdsnccns.c,260,"NCSErrorMessage") CT/DS RPC Error: DSR040 - lb_lookup_annotation
abnormal return
(4CBE1EB2.0427-FE8:kdsnccns.c,59,"ConvertNCSStatus") NCS Status Code: 1c020006
(4CBE1EB2.0428-FE8:kdsnccns.c,209,"ConvertNCSStatus") CT/DS status returned: 172
(4CBE1EB2.0429-FE8:ko4locbr.cpp,594,"locMgr:lbLookupHub") Error <5> returned from SQL1_lbLookup for directAddr
<@IP.PIPE:SUGRAMIHQMONE01.INFO53.COM>
(4CBE1EB2.042A-FE8:ko4locbr.cpp,424,"locMgr:locateEverybody") lblookupHub returned error <5> for directAddr
<$MHM:@IP.PIPE:SUGRAMIHQMONE01.INFO53.COM>
(4CBE1EB2.042B-FE8:ko4ib.cpp,1964,"IBInterface_directConnect") location broker failure - error <5>
(4CBE1EB1.0502-FE8:kdcc1sr.c,460,"rpc_sar") Connection failure: "ip.pipe:#10.224.194.215:1918",
1C010001:1DE00045, 1, 5(2), FFFF/2, d65200:1.1.1.10, tms_ctbs622mdv.d9268a
(4CBE1EB1.0503-FE8:kdcl0cl.c,142,"KDCL0_ClientLookup") status=1c020006, "location server unavailable",
ncs/KDC1_STC_SERVER_UNAVAILABLE
```
- Solution: Delete offline entries and restart the Universal Agent

7

Reading agent logs, Part 3 - Examples, solutions, and more

© 2013 IBM Corporation

The keywords **unavailable**, **abnormal**, and **fail** help you to locate a communication problem. Take a moment and look at the highlighted words in the example log file. Do you see these errors?

- **abnormal** return
- Connection **failure**
- location server **unavailable**

Example: Installation aborts

- Problem Description: Windows® Agent Install Issue - String Access Out of Bounds
- Component: Windows OS agent (NT)
- Log name: Abort_IBM_Tivoli_Monitoring_#####_####.log
- Example:


```
8-23-2012 10:10:20 RegistryGetKeyValueXX Key(HKLM\SYSTEM\CurrentControlSet\Control\Session
Manager\Environment) Value(Path) ERROR:The system cannot find the file specified.
8-23-2012 10:10:22 OnAbort Entered
8-23-2012 10:10:22 CallDllFixAndLog CALLBACK_OnAbort(0, E:\WINDOWS_base621\WINDOWS_Base-
>C:\Tivoli\ITM6)
8-23-2012 10:10:22 SetEnvironmentVariable: ITMINSTALL_DBG_LVL value DEBUG_MAX set by
SetEnvironmentVariableA
8-23-2012 10:10:22 SetEnvironmentVariable: ITMINSTALL_DBG_OUT value
C:\Users\ADMINI~1\AppData\Local\Temp\2\{CB2E0CF4-F87B-4CFB-9167-67AE03B7C74F}\IBM Tivoli Monitoring
20120823 1009.log set by SetEnvironmentVariableA 8-23-2012 10:10:22 CITMISUtilApp::InitInstance
8-23-2012 10:10:22 CALLBACK_OnAbort - Entry
8-23-2012 10:10:22 CALLBACK_OnAbort - Exit[0]
8-23-2012 10:10:22 CITMISUtilApp::ExitInstance
8-23-2012 10:10:22 UTiIDll:CALLBACK_OnAbort returned code 0
8-23-2012 10:10:22 RegistryGetKeyValue Key(HKLM\SYSTEM\CurrentControlSet\Control\Terminal Server)
Value(TSEnabled) ERROR: The system cannot find the file specified.
```
- Solution: There is a missing or corrupted path statement on the target server

In this example, the keyword **abort** helps to locate a registry error that is traced to a missing path file.

Notice the bolded error message near the end of the slide that begins with **RegistryGetKeyValue**. The log entry ends with **ERROR:The system cannot find the file specified**. Hence, there is a missing file path.

Example: installation fails and cores

- Problem Description: 6.2.2 UNIX® Agent fails to install and cores
- Component: UNIX OS agent (UX)
- Log name: <ITM HOME>/logs/candle_installation.log
- Example:

```
runGSKit: error Return error code: 99 runGSKit: error GSKit check failure, script:  
/opt/IBM/ITM/sol286/gs/bin/private_verifyinstall_64 runGSKit: Removing gskit ver file and  
gskit component from /opt/IBM/ITM runGSKit: error sol286 - GSK check error,  
verifyInstall test failed
```
- Solution: To resolve these gskit errors encountered during a KUX install on Solaris, the customer upgraded their hardware and OS

In this example, the keyword **fail** helps to locate a GSKit script problem during a UNIX agent installation. Do you see the bold text in the message for the word **fail**?

Example: historical data corruption

- Problem Description: Corrupted History for NTMemory and NTProcessor
- Component: Windows OS agent (NT)
- Log name: <hostname>_nt_kntcma_<timestamp>-0#.log
- Example:


```
"CTHistorySource::validateRow") History file
E:\IBM\ITM\TMAITM6\logs\History\KNT\NTPROCSSR corruption found at file position
00000000. EndOfFileReached = 0, CurrRowValid = 0, NextRowValid = 0.
(4F2542F2.000C-1010:khdxhist.cpp,3334,"CTHistorySource::validateRow")
<0x2911FE8,0x1B8> Corrupted row: +4F2542F2.000C 00000000 00000000 00000000
00000000 00000000 .....
+4F2542F2.000C ***** 00000010 - 000001B0: Same as above *****
+4F2542F2.000C 000001B0 00000000 00000000 .....
(4F2542F2.000D-1010:khdxhist.cpp,3664,"readAheadRow") History file
<E:\IBM\ITM\TMAITM6\logs\History\KNT\NTPROCSSR> corruption recovery failed, retries
= 0. Unknown number of records dropped.
(4F2542F2.000E-1010:khdxdacl.cpp,2033,"routeData") Found 1 rows corrupted data for
NTPROCSSR. Rows were ignored
```
- Solution: Do not configure two or more agents to write to the same historical directory

10

Reading agent logs, Part 3 - Examples, solutions, and more

© 2013 IBM Corporation

In this example, the keyword **corrupt** helps to identify a configuration problem where two agents are configured to write to the same historical directory. Do you see the word **corrupt** in the message? Notice the search is not case sensitive.

Example: Missing capacity ranges

- Problem description: UPS capacity ranges are not reading values
- Component: APC InfraStruXure Agent (E7)
- Log name: <hostname>_ e7_<instance>_ke7agent_<timestamp>-0#.log
- Example:

```
ERROR MESSAGE: "Unable to open Metafile
"C:\IBM\ITM\TMAITM6\logs\History\KE7\test\KE7AIRIR11.hdr" " ERROR MESSAGE:
"Unable to open Metafile "C:\IBM\ITM\TMAITM6\logs\History\KE7\test\KE7AIRIRS0.hdr" "
ERROR MESSAGE: "Unable to open Metafile
"C:\IBM\ITM\TMAITM6\logs\History\KE7\test\KE7AIRFMSY.hdr" " ERROR MESSAGE:
"Unable to open Metafile "C:\IBM\ITM\TMAITM6\logs\History\KE7\test\KE7AIRIR20.hdr" "
ERROR MESSAGE: "Unable to open Metafile
<C:\IBM\ITM\TMAITM6\logs\History\KE7\test\KE7AIRIRG6.hdr>" (4DD52551.0000-
8D0:snmpqueryclass.cpp,1265,"handle_snmp_response_async")ERROR: decoded PDU is
null -- this is a timeout scenario
```
- Solution: Software defect, APAR opened

In this example, the keywords **unable** and **timeout** help to locate the cause of missing Tivoli Enterprise Portal information from the E7 agent.

This PMR was caused by a software defect and an APAR lead to a change that corrected a problem with the agent code.

Example: UX agent crash with SIGSEGV and SIGILL

- Problem Description: core dump on AIX® 6.2.3FP1 agent on AIX 7.1
 - Component: UNIX OS agent (UX)
 - Log names: <hostname>_ux_#####.log, <hostname>_ux_aixdp_daemon_<timestamp>-0#.log, <hostname>_ux_mount_stat_<timestamp>-0#.log, <hostname>_ux_kuxagent_<timestamp>-0#.log, <hostname>_ux_kux_vmstat_<timestamp>-0#.log, <hostname>_ux_stat_daemon_<timestamp>-0#.log, <hostname>_ux_kcawd_<timestamp>-0#.log
 - Example:


```
**** Fatal Error (4) Detected in kuxagent or a helper binary ****
**** Stacktrace in standard logs. Enable KBB_SIG1=dumpoff in ini for core dumps ****
+++PARALLEL TOOLS CONSORTIUM LIGHTWEIGHT COREFILE FORMAT version 1.0
+++LCB 1.0 Mon Mar 26 13:56:49 2012 Generated by IBM AIX 7.1
+++ID Node 0 Process 7078194 Thread 16
****FAULT "SIGILL - Illegal instruction"
+++STACK
**** Stacktrace in standard logs. Enable KBB_SIG1=dumpoff in ini for core dumps ****
+++++
**** Fatal Error (11) Detected in kuxagent or a helper binary ****
**** Stacktrace in standard logs. Enable KBB_SIG1=dumpoff in ini for core dumps ****
+++PARALLEL TOOLS CONSORTIUM LIGHTWEIGHT COREFILE FORMAT version 1.0
+++LCB 1.0 Mon Mar 26 13:56:49 2012 Generated by IBM AIX 7.1
#
+++ID Node 0 Process 7078194 Thread 2
****FAULT "SIGSEGV - Segmentation violation"
+++STACK
pthread_mutex_init : 0x000000c4
+++++
(4F85F1D5.0086-1.aixtranslator.c.2693,"ux_initializeCollectors") Spmilnit error: [no msg] attempt:1
(in <hostname>_ux_aixdp_daemon_<timestamp>-0#.log only)
```
- Solution: Signal and SPML errors required the AIX perfagent.tools fileset to be upgraded

In this example, you can see how the keywords **fatal** and **fault** along with the SIGSEGV and SIGILL signal errors are in the various logs shown. This information indicates the need to upgrade the AIX **perfagent.tools** fileset.

Notice how the errors are present in the agent logs listed.

Example: Customer script terminates framework

- Problem Description: UNIX Log agent process does not autostart
- Component: UNIX Log Agent (UL)
- Log name: <hostname>_ul_kulagent_<timestamp>-0#.log
- Example:


```
(4F3136AA.0000-3:logmanager.cpp,833,"waitForSignal") Stop (SIGTERM) signal received.
(4F3136AA.0001-3:logmanager.cpp,854,"waitForSignal") Exit: 0x0
(4F3136AA.0002-3:logmanager.cpp,1769,"managerThread") Log manager terminating framework.
(4F3136AA.0ADC-3:kraafmgr.cpp,2423,"TEMA_Termination") TEMA_ Termination exiting
(4F3136AA.0ADD-3:kraafmgr.cpp,2424,"TEMA_Termination") Exit
(4F3136AA.0AFC-1:logmanager.cpp,472,"stopManagerThread") Error canceling manager thread. Return
code = 3; No such process
(4F3136AA.0AFD-1:logmanager.cpp,485,"stopManagerThread") Manager thread terminated.
(4F3136AA.0AFE-1:logmanager.cpp,496,"stopManagerThread") Log manager mutex destroyed.
(4F3136AA.0AFF-1:logmanager.cpp,507,"stopManagerThread") 'inuse' condition variable destroyed.
(4F3136AA.0B00-1:logmanager.cpp,518,"stopManagerThread") 'server-up' condition variable destroyed.
(4F3136AA.0B01-1:logmanager.cpp,521,"stopManagerThread") Exit (4F3136AC.0023-
4:logfile.cpp,1394,"LogThread") Thread for log file /var/log/secure terminating.
```
- Solution: This problem was traced to the script the customer calls. The script does not run from the CLI. The customer corrected the script problem

13

Reading agent logs, Part 3 - Examples, solutions, and more

© 2013 IBM Corporation

This example shows how the keyword **terminat** without the letter "e" can be used to detect the words **Termination**, **terminated**, and **terminating**.

Often Tivoli Monitoring agent logs detect problems that do not have an agent problem as the root cause.

Example: Sub-daemon expires for Linux® OS agent

- Problem Description: zLinux agent cores
- Component: Linux OS agent (LZ)
- Log name: <hostname>_lz_klzagent_<timestamp>-0#.log
- Example: Messages like these were repeated through out the klxzagent logs
(4E565353.0000-E:filestats.cpp,300,"executeStatfsInSeparateThread") WARNING: The statfs timeout **expired!**
(4E565353.0001-E:filestats.cpp,301,"executeStatfsInSeparateThread") WARNING: The mounted file system "/home" is probably unreachable
(4E565353.0002-E:filestats.cpp,112,"GetFileStats") statfs timed out for /home
(4E565353.0003-E:filestats.cpp,112,"GetFileStats") statfs timed out for /home
(4E565353.0004-E:filestats.cpp,112,"GetFileStats") statfs timed out for /home
(4E5653C8.0000-E:filestats.cpp,300,"executeStatfsInSeparateThread") WARNING: The statfs timeout **expired!**
- Solution: This problem stopped occurring while troubleshooting. Customer did not want to pursue

In this example, the keyword **expire** helps to detect a timed out message. When you search for **expire**, you also find the word **expired**.

Occasionally you can use the agent logs to identify errors that, for whatever reason, the customer chooses to not pursue.

Example: Tivoli Enterprise Portal shows some agent columns as unavailable

- Problem Description: Unavailable being displayed for some columns of vfilers
- Component: NetApp Storage Agent (NU)
- Log name: <hostname>_ nu_ <instance>_knuagent_<timestamp>-0#.log
- Example:
 - (4F4212BB.0006-1D28:khdthrd.cpp,122,"CTThread") pthread_attr_setstacksize with Stack Size 524288 ended with Error -1.
 - (4F4212C7.0000-1D28:utilities.cpp,205,"parseNumericString") Invalid characters **unavailable** found getting numeric value from **unavailable**, returning 0.000000
- Solution: This customer encountered a known APAR
 - IBM Tivoli Monitoring for Virtual Servers: NetApp 7.1.0-TIV-ITM_NETAPP-IF0001
 - <http://www-01.ibm.com/support/docview.wss?uid=swg24032275>

Here the keyword **unavailable** points to a known APAR.

In this example, the error reported by the customer, the search word that is used to locate the problem, and solution all contained the word **unavailable**.

Example: Failed to start an unavailable agent (1 of 2)

- **Problem Description:** Agent Management Services will not restart agent instance if cinfo-r returns a "process not running" status for that instance
- **Components:** Agent for PeopleSoft Enterprise Application Domain (P8) and Linux OS agent (LZ)
- **Log name:** <hostname>_lz_klzagent_<timestamp>-0#.log (Note: no errors were found in <hostname>_ux_kcawd_<timestamp>-0#.log)
- **Example:**

```
(4F0649C9.0000-4:kcawd.cpp,290,"checkAvailability") Starting agent KP8062100000 for the first time.
(4F0649C9.0001-F:kbbssge.c,52,"BSS1_GetEnv") USER="root"
(4F064AF5.0000-4:kcawd.cpp,317,"checkAvailability") Availability script failed and agent KP8062100000 is down,
rc = 1, command output < /opt/IBM/ITM +4F064AF5.0000 p8 Command STATUS failed +4F064AF5.0000 >.
(4F064AF5.0001-4:kcawd.cpp,318,"checkAvailability") Starting unavailable agent KP8062100000.
(4F064AF5.0002-5:klz31agt.cpp,175,"PAS_AlertListener::AlertMe()") originnode="hostname:LZ"
(4F064AF5.0003-5:klz31agt.cpp,176,"PAS_AlertListener::AlertMe()") timestamp="1120105181429000"
(4F064AF5.0004-5:klz31agt.cpp,177,"PAS_AlertListener::AlertMe()") pasagname="Monitoring Agent for
PeopleSoft Domains"
(4F064AF5.0005-5:klz31agt.cpp,178,"PAS_AlertListener::AlertMe()") status=2 (4F064AF5.0006-
5:klz31agt.cpp,179,"PAS_AlertListener::AlertMe()") procname="kp8agent" (4F064C21.0000-
4:kcawd.cpp,317,"checkAvailability") Availability script failed and agent KP8062100000 is down, rc = 1, command
output < /opt/IBM/ITM +4F064C21.0000 p8 Command STATUS failed Solution:
(Solution on next screen)
```

In this example, the keywords **unavailable** and **fail** help to identify an inconsistency in the code where the agent cannot be restarted even though the log indicates that the agent is stopped.

This example continues on the next screen with the example messages repeated and the solution.

Example: Failed to start an unavailable agent (2 of 2)

- Example:

```
(4F0649C9.0000-4:kcawd.cpp,290,"checkAvailability") Starting agent KP8062100000 for the first time.
(4F0649C9.0001-F:kbssge.c,52,"BSS1_GetEnv") USER="root"
(4F064AF5.0000-4:kcawd.cpp,317,"checkAvailability") Availability script failed and agent KP8062100000
is down, rc = 1, command output < /opt/IBM/ITM +4F064AF5.0000 p8 Command STATUS failed
+4F064AF5.0000 >. (4F064AF5.0001-4:kcawd.cpp,318,"checkAvailability") Starting unavailable agent
KP8062100000. (4F064AF5.0002-5:klz31agt.cpp,175,"PAS_AlertListener::AlertMe()")
originnode="hostname:LZ"
(4F064AF5.0003-5:klz31agt.cpp,176,"PAS_AlertListener::AlertMe()") timestamp="1120105181429000"
(4F064AF5.0004-5:klz31agt.cpp,177,"PAS_AlertListener::AlertMe()") pasagname="Monitoring Agent for
PeopleSoft Domains"
(4F064AF5.0005-5:klz31agt.cpp,178,"PAS_AlertListener::AlertMe()") status=2 (4F064AF5.0006-
5:klz31agt.cpp,179,"PAS_AlertListener::AlertMe()") procname="kp8agent" (4F064C21.0000-
4:kcawd.cpp,317,"checkAvailability") Availability script failed and agent KP8062100000 is down, rc = 1,
command output < /opt/IBM/ITM +4F064C21.0000 p8 Command STATUS failed Solution:
```

- Solution: Setting trace to KBB_RAS1: ERROR(UNIT:KCA ALL) revealed multiple instances running
 - +4FA004C4.0007 KCIIN0208E Instance <instance name> already running.
 - +4FA004C4.0007 KCIIN0195E Agent already started...

Problem resolved in OS Agent 6.2.2.4-TIV-ITM_UNIX-IF0008

In this example, the support staff requested more tracing and then could detect the problem.

The errors near the beginning of the example, indicate a need to increase the trace level to **KBB_RAS1: ERROR(UNIT:KCA ALL)**.

This new trace revealed multiple agent instances are running concurrently.

Example: SNMP pull timeout

- Problem Description: Agentless for Solaris 6.22.06 SNMP pull causing false situations to fire
- Component: Agentless Monitoring for Solaris Operating Systems (KR6)
- Log name: <hostname>_r6_<instance>_kr6agent_<timestamp>-0#.log
- Example:


```
(2011/11/09,22:52:22.000A-A:snmpqueryclass.cpp,1288,"handle_snmp_response_async") ERROR:
decoded PDU is null -- this is a timeout scenario
(2011/11/09,22:52:22.000B-28:snmpqueryclass.cpp,1956,"internalCollectData") Timeout occurred. No
response from agent *****.org.
(2011/11/09,22:52:40.0000-3:configdiscoveryqueryclass.cpp,250,"internalCollectData") Removing R6:
*****:SMA node, no valid data collection.
(2011/11/09,22:52:40.0001-3:subnodemanager.cpp,1060,"SubnodeManager::removeSubnode") Leaving
subnode - allow it to continue to try to collect data for MSN "R6: *****:SMA"
(2011/11/09,22:53:10.0000-3:configdiscoveryqueryclass.cpp,250,"internalCollectData") Removing
R6:*****:SMA node, no valid data collection.
(2011/11/09,23:23:29.0000-C:snmpqueryclass.cpp,1931,"internalCollectData") Timeout occurred. No
response from agent
```
- Solution: This problem is not caused by an agent. The agent works as designed. The servers could not pull process detail because they are virtual servers.

Here you can see the keyword **timeout** is used to identify a problem with the way a customer uses Agentless Monitoring for Solaris.

In this PMR, the agent works as designed.

Example: Watchdog timeout, multiple copies of OS agents

- Problem Description: Linux OS agent starts multiple processes
- Component: Linux OS agent (LZ)
- Log name: hostname>_ux_kcawd_<timestamp>-0#.log
- Example:

```
(4F029AC1.0000-3:kcacmdunx.cpp,332,"executeCmd") Command did not finish within timeout - errno = 999999
(4F029AC1.0001-3:kcacmdunx.cpp,211,"executeCmd") Error: kca_popen failed, errno = 11
(4F029AC1.0002-3:kcawd.cpp,246,"checkAvailability") Availability script timeout/failed but agent klzagent is still up.
(4F029AC1.0003-3:kcawd.cpp,247,"checkAvailability") Restarting unhealthy agent klzagent.
(4F029AC1.0004-3:kcacmdunx.cpp,211,"executeCmd") Error: kca_popen failed, errno = 11
(4F029AC1.0005-3:kcawd.cpp,246,"checkAvailability") Availability script timeout/failed but agent klzagent is still up.
(4F029AC1.0006-3:kcawd.cpp,247,"checkAvailability") Restarting unhealthy agent klzagent.
(4F029AC1.0007-3:kcaunx.cpp,586,"DisableWatchdogStop") Creating Watchdog disable stop file /opt/IBM/ITM/ix8266/lz/bin/.kcawd_disable_stop.
(4F029AC1.0008-3:kcactrl.cpp,3726,"startDetachedThread") ERROR: pthread_create() failed (rc 11)
(4F029AC1.0009-3:kcactrl.cpp,3993,"cmdExecutionThread") command incomplete-- Error starting thread.
```

- Solution: This problem is corrected in *IBM Tivoli Monitoring 6.2.2 Fix Pack 6 (6.2.2-TIV-ITM-FP0006)*

The keywords **timeout** and **fail** help to identify a Proxy Agent Services (watchdog) problem that started multiple copies of the Linux OS agent.

Example: Connection failure to the virtual center

- Problem Description: Connection Failure to Virtual Center
- Component: VMWare VI Agent (VM)
- Log name: kvm_data_provider_<instance>_startup.log and kvm_data_provider_<instance>_#.log
- Example:

```
2012-08-27 15:52:26 0 WARNING: Connection.open: Caught an exception while cleaning up a failed connection attempt.
2012-08-27 15:52:26 0 SEVERE: DataSource.start: Failed to connect to: *****.com
2012-08-27 15:52:26 0 SEVERE: DataProvider.addDataSource: Initial attempt to connect to *****.com.ibm.tivoli.monitoring.agent.kvm.vmware.DataSource@54175417 failed.
2012-08-27 15:52:48 1 INFO: Connection.open: Using jar local WSDL files.
2012-08-27 15:52:52 1 WARNING: Connection.open: Caught an exception while cleaning up a failed connection attempt.
2012-08-27 15:52:52 1 SEVERE: DataSource.start: Failed to connect to: *****.com
2012-08-27 15:52:52 1 WARNING: DataProvider.run: Failed to start datasource *****.com
```
- Solution: The customer uses a not valid account and password to connect to the virtual center

This slide shows how the keywords **exception**, **severe**, and **fail** helped to locate a not valid account and password problem that the customer used to access their virtual center.

Example: datastores msn is unavailable

- Problem Description: VMWare VI - datastores msn is unavailable
- Component: VMWare VI Agent (VM)
- Log names: kvm_data_provider_<instance>_#.log and knu_data_provider_<instance>_#.log
- Example:
 - The kvm_data_provider_<instance>_#.log shows these errors
2012-01-18 09:36:38 11 SEVERE: PerformanceMetricCollectionUnit.updateCache: Could not get a PerfQuerySpec to gather performance data for (VirtualMachine) vm-8806: db24.ipvn01. **Aborting.** requested by (VMDatastoreUtilizationAttributeGroup).
2012-01-19 08:54:13 20 SEVERE: FactoryInterfaCEController.processCollectDataRequest: Caught an **exception** while sending data for attribute group Server. Aborting.
java.lang.Object: Error writing to socket
 - The knu_data_provider_<instance>_#.log shows these errors
2012-05-08 07:46:44 14 SEVERE: Inventory.getAvailablePerformanCECounters: Invocation of perf-object-counter-list-info failed: netapp.manage.NaAPIFailed**Exception**: There is no controller named '636'. (errno=13001). **Aborting**
 - Solution: The Volume Name attribute of KNU Volumes attribute group has a 64 character limit. The NetApp_Volume_Name attribute of KVM Datastores attribute group is limited to 100 characters

The keywords **abort** and **exception** help to identify a software limitation that keeps the agent information from displaying in the Tivoli Enterprise Portal.

To resolve this PMR, IBM created a technote to make resolving this problem easier for someone who finds it.

Example: Duplicate ESX servers shown

- Problem Description: Duplicate ESX servers shown
- Component: VMWare VI Agent (VM)
- Log name: kvm_data_provider_<instance>_#.log
- Example:

```
2012-08-27 12:25:34 10 SEVERE: Connection.open: Caught a RemoteException while
attempting to open a connection to http://<ip address>:80/sdk

org.apache.axis.AxisFault: (503)Service Unavailable

Stack trace:

2012-08-27 12:35:45 26 WARNING: DataProvider.generateSubnodeKey: Subnode key
generated for host system <hostname> conflicts with an existing key. Using <hostname>_2
instead. Note that this might cause inconsistencies with historical data storage
```
- Solution: The ESX host addresses are not unique within 25 characters. This problem can typically be resolved by shortening the agent instance name

Here is an example of how the keywords **exception**, **unavailable**, **fault**, and **conflict** are used to identify a configuration limitation.

In this example, the ESX host addresses are not unique within 25 characters.

This problem can typically be resolved by shortening the agent instance name.

Example: UNIX OS Agent crash

- Problem Description: UNIX OS Agent crash
- Component: UNIX OS Agent (UX) v 6.23
- Log name: <hostname>_ux_aixdp_daemon_<timestamp>-0#.log
- Example:

```
(4FEB175D.0004-1:aixtranslator.c,2691,"ux_InitializeCollectors") Spmilnit error: Spmi:
Unable to remove the unused Shared Area ( shmctl, error 1 ), attempt:1
(4FEB1762.0000-1:aixtranslator.c,2691,"ux_InitializeCollectors") Spmilnit error: Spmi:
Unable to remove the unused Shared Area ( shmctl, error 1 ), attempt:2
(4FEB1767.0000-1:aixtranslator.c,2691,"ux_InitializeCollectors") Spmilnit error: Spmi:
Unable to remove the unused Shared Area ( shmctl, error 1 ), attempt:3
```
- Solution: Add the line `KUX_AIXDP=false` in the `ux.ini` to prevent the `aixdp_daemon` dataprovider from starting and allow the rest of the UX agent to run

This error repeats itself multiple times in the log shown. The keyword **unable** helps to locate problem.

To work around the problem, add the line **KUX_AIXDP=false** in the **ux.ini** file to prevent the **aixdp_daemon** dataprovider from starting and allow the rest of the UX agent to run properly.

This problem was resolved in 6.2.3-TIV-ITM-FP0002.

Example: UNIX OS agent Fatal Error (10)

- Problem Description: kuxagent crashes with Fatal Error (10). No core generated
- Component: UNIX OS agent (UX)
- Log name: <hostname>_ux_#####.log
- Example:
(4C650B99.0000-1:stacktrace.cpp,100,"exit_with_notification") **FATAL ERROR: Fatal error (10) detected. Shutdown initiated. (4C650B99.0001-2:signalmanager.cpp,474,"managerThread") Termination signal received. Exiting... . Signal 10 is SIGBUS**
- Solution: This is a non-specific defect. Level 3 support assisted to resolve the problem

The keywords **fatal**, **sigbus**, and **terminat** without the letter "e" are used to locate a problem.

The problem resolution is to upgrade an older agent.

Example: z/Linux agent memory/CPU spikes

- Problem Description: z/Linux agent memory/CPU spikes
- Component: zLinux OS Agent (LZ)
- Log name: <hostname>_lz_klzagent_<timestamp>-0#.log
- Example:

```
(4CC09BDE.0007-8:klzmain.cpp,136,"CleanUp") Signal captured:  
SIG1_SIGILL(4CC09BDE.0008-8:klzmain.cpp,137,"CleanUp") Shutting down agent...  
(4CC09BDE.0009-9:khdxppts.cpp,1537,"clientExportThread") Exiting function  
clientExportThread, quit: 1, requests on queue: 0 (4CC09BDE.000A-  
8:klzmain.cpp,142,"CleanUp") Issuing Exit...  
(4CC09BDE.000B-8:klzmain.cpp,136,"CleanUp") Signal captured: SIG1_SIGILL  
(4CC09BDE.000C-8:klzmain.cpp,137,"CleanUp") Shutting down agent...  
(4CC09BDE.000D-8:pashandlers.cpp,138,"Unregister_PASHandlers") Failed to deregister  
alert listener.
```
- Solution: This is a non-specific defect. Level 3 support assisted to resolve the problem

In this example, the signal error message **SIG1_SIGILL** helps to identify the cause of intermittent CPU spike and memory problems.

Example: i5 OS agent is not responding

- Problem Description: i5 OS agent is not responding
- Component: AS/400® Agent i5 OS (A4)
- Log name: QAUTOTMP/KA4AGENT01
- Example:


```
(4F414B54.0000-5:kdei0af.c,113,"from_name") Status
1DE00003=KDE1_STC_BADDRAWNAME (4F414B54.0001-
5:kdck1fn.c,76,"socket__from_name") status=10020003, "invalid string address format",
ncs/KDC1_STC_ BAD_NUMERIC_NAME
(4F414B54.0002-5:kraarreg.c,2071,"PrimaryTEMSperiodicLookupThread") Unable to create
lookup socket to host IP.PIPE:*non175.7.149, status = 10020003
(4F415CE8.0000-5:kdei0af.c,113,"from_name") Status
1DE00003=KDE1_STC_BADDRAWNAME (4F415CE9.0000-
5:kdck1fn.c,76,"socket__from_name") status=10020003, "invalid string address format",
ncs/KDC1_STC_ BAD_NUMERIC_NAME
(4F415CE9.0001-5:kraarreg.c,2071,"PrimaryTEMSperiodicLookupThread") Unable to
create lookup socket to host IP.PIPE:*non175.7.149, status = 10020003
```
- Solution: Corrected by reconfiguring the primary Tivoli Enterprise Monitoring Server address

In this example, the keyword **unable** helps to identify extraneous characters in the IP address that is used to configure the agent to communicate with its Tivoli Enterprise Monitoring Server.

To resolve the problem, reconfiguring primary Tivoli Enterprise Monitoring Server address.

Example: CEC agent, RSiErrno = 280, Unmonitored LPAR

- Problem Description: CEC agent cannot contact host
- Component: System p® CEC Base agent (PK)
- Log name: itmProvider.trc
- Example:


```
+47186 TIMEL Error: Mon Jan 31 22:19:05 2011 Unable to Open LPAR-third attempt
<customer_url1.com>
+47187 DEBUG: RSiErrno = 280 *** Failed to open LPAR <name>.
+48281 TIMEL Error: Mon Jan 31 22:19:20 2011 Unable to Open LPAR-third attempt
<customer_url2.com>
+48282 DEBUG: RSiErrno = 280
+48339 TIMEL Error: Mon Jan 31 22:19:36 2011 Unable to Open LPAR-third attempt
<customer_url3.com>
+48340 DEBUG: RSiErrno = 280
```
- Solution: 3 LPARs that were showing "unmonitored." After comparing the working and non-working systems, it was determined that the customer was encountering a known problem with multi-homed systems. After forcing xmtopas traffic on to specified network address, the unmonitored LPAR started responding.

27

Reading agent logs, Part 3 - Examples, solutions, and more

© 2013 IBM Corporation

Here you can see how the keywords **fail** and **unable** are used to locate **RSiErrno=280** messages.

RSiErrno=280 implies that the CEC agent attempted communicating with the LPAR, but did not receive a response.

After the CEC agent gets a list of LPAR from the HMC, it attempts to communicate with each of the LPAR to get metrics from the individual LPAR.

Because the CEC agent communicates through the RSi protocol, the UDP port 2279 must be open in both directions.

The first step is to make an RsiOpen call to the end LPAR.

If the RsiOpen call does not succeed or if the LPAR at the other end fails to respond, the LPAR is marked as **unmonitored**.

Recommendation: Scan System p logs for **RSiErrno = 280** and **RSiErrno = 288** messages.

Example: CEC Agent cores RSiErrno = 288

- Problem Description: CEC Agent cores
- Component: System p CEC Base agent (PK)
- Log name: <hostname>_pk_CECDataProvider_<timestamp>-0#.log
- Example:


```
(4FD1A328.007C-3:CECmodule.c,936,"partition_data_feed") TIMEL Fri Jun 8 03:00:56 2012 Calling
open_LPAR: hostname=
(4FD1A328.007D-3:CECmodule.c,84,"open_LPAR") Entry
(4FD1A328.007E-3:CECmodule.c,110,"open_LPAR") open_LPAR: for index 55 <ga016a4db>
(4FD1A328.007F-3:CECmodule.c,116,"open_LPAR") TIMEL: Fri Jun 8 03:00:56 2012 Before OPEN for
(4FD1A328.0080-3:CECmodule.c,117,"open_LPAR") Attempting RSiOpen for rh_index: 50
(4FD1A328.0081-3:CECmodule.c,132,"open_LPAR") TIMEL Error: Fri Jun 8 03:00:56 2012 Unable to
Open LPAR-first attempt <>
(4FD1A328.0082-3:CECmodule.c,133,"open_LPAR") DEBUG: RSiErrno = 288
(4FD1A328.0083-3:CECmodule.c,140,"open_LPAR") TIMEL Error: Fri Jun 8 03:00:56 2012 Unable to
Open LPAR-second attempt <>
(4FD1A328.0084-3:CECmodule.c,141,"open_LPAR") DEBUG: RSiErrno = 288
```
- Solution: After analyzing the snapcore, this problem was traced to a known APAR

The keyword **unable** in this example helps to locate RSiErrno=288.

RSiErrno=288 indicates a wrong parameter that is passed.

Notice that the full hostname is missing. You can see in the in the second line of the example **hostname =**). The blank indicates there is no name.

Conclusion

- Finding errors or the apparent problem might or might not resolve the issue
- How quickly the problem can be resolved depends on if this issue was previously encountered and whether a previous solution is provided
- Use search tools like Google or Ask.com to obtain more information
- Often IBM APARs (Authorized Program Analysis Reports) include the symptoms and error messages in the text published to the web
- Errors that appear in Tivoli agent logs might be rooted in the products IBM supports and not necessarily the agent that reported the problem
- Companies like Microsoft and Sun often provide solutions on their websites that can be used to resolve apparent agent problems
- Maintain a variety of current IBM manuals and User's Guides on your local drive for the products you work with

Finding errors or the apparent problem might or might not resolve the issue.

How quickly the problem can be resolved depends on whether this issue was previously encountered and whether a previous solution is provided.

After you review the log files and find potential errors, symptoms, and messages, you can use search tools like Google or Bing to obtain more information.

Google provides a surprising number of matches that point to IBM Developer Works and other vendor forums where problems are reported and solved.

Google also helps cross-reference problems. Some errors that are associated with Tivoli agents might be rooted in the products that the agents monitor, not necessarily the agent that reported the problem.

Companies like Microsoft and Sun often provide solutions on their websites that you can use to resolve an apparent agent problem.

Finally, it always helps to have a wide variety of current IBM manuals and User Guides available on your local drive for the products you work with.

The search capabilities that are provided locally by Adobe Reader can be superior to any web-based tool for reading PDF files.

IBM support is always willing and able to help resolve your problems.

These presentations are intended to provide the customer with insights into the inner workings of our products and to help reduce the time required to resolve some of the

problems you may encounter.

Summary

Now that you have completed this module, you can perform these tasks:

- Describe how error messages and keywords appear in logs
- Identify possible solutions for the errors found
- Describe how to proceed once a suspected problem is identified

Now that you completed this module, you can perform these tasks:

- Describe how error messages and keywords appear in logs
- Identify possible solutions for the errors found
- Describe how to proceed once a suspected problem is identified

Other helpful references (1 of 2)

- Various versions of Tivoli Monitoring manuals from V5 to V6.23 in the Tivoli Monitoring documentation:

http://pic.dhe.ibm.com/infocenter/tivihelp/v15r1/index.jsp?topic=/com.ibm.itm.doc_6.2.2fp2/welcome.htm

- 6.23 Fix Pack 1 troubleshooting

http://pic.dhe.ibm.com/infocenter/tivihelp/v15r1/topic/com.ibm.itm.doc_6.2.3fp1/itm623fp1_troubleshoot.pdf

- 6.23 Fix Pack 1 message guide

http://pic.dhe.ibm.com/infocenter/tivihelp/v15r1/topic/com.ibm.itm.doc_6.2.3fp1/itm623fp1_messages.pdf

- 6.22 Fix Pack 2 troubleshooting

http://pic.dhe.ibm.com/infocenter/tivihelp/v15r1/topic/com.ibm.itm.doc_6.2.2fp2/itm_troubleshoot.pdf

- 6.22 message guide

http://pic.dhe.ibm.com/infocenter/tivihelp/v15r1/topic/com.ibm.itm.doc_6.2.2/itm_messages.pdf

This first reference page contains IBM links to helpful websites and PDF files.

Other helpful references (2 of 2)

- For a comprehensive list of product codes see IBM Tivoli Monitoring ITM 6.X Product Codes
 - <http://www-01.ibm.com/support/docview.wss?uid=swg21265222>
- See AIX Signals signification for more information on AIX Signals
 - <http://www-01.ibm.com/support/docview.wss?uid=swg21145669>

See the Dan website for a tool to convert the hexadecimal time stamp in logs to reader friendly formats

- <http://dan.drydog.com/unixdatetime.html>

This second reference page contains IBM links to helpful websites and two links to non IBM websites.

Trademarks, disclaimer, and copyright information

IBM, the IBM logo, ibm.com, AIX, AIX 6, AS/400, System p, and Tivoli are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of other IBM trademarks is available on the web at "[Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml)" at <http://www.ibm.com/legal/copytrade.shtml>

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

Microsoft, Windows, and the Windows logo are registered trademarks of Microsoft Corporation 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.

THE INFORMATION CONTAINED IN THIS PRESENTATION IS PROVIDED FOR INFORMATIONAL PURPOSES ONLY. WHILE EFFORTS WERE MADE TO VERIFY THE COMPLETENESS AND ACCURACY OF THE INFORMATION CONTAINED IN THIS PRESENTATION, IT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. IN ADDITION, THIS INFORMATION IS BASED ON IBM'S CURRENT PRODUCT PLANS AND STRATEGY, WHICH ARE SUBJECT TO CHANGE BY IBM WITHOUT NOTICE. IBM SHALL NOT BE RESPONSIBLE FOR ANY DAMAGES ARISING OUT OF THE USE OF, OR OTHERWISE RELATED TO, THIS PRESENTATION OR ANY OTHER DOCUMENTATION. NOTHING CONTAINED IN THIS PRESENTATION IS INTENDED TO, NOR SHALL HAVE THE EFFECT OF, CREATING ANY WARRANTIES OR REPRESENTATIONS FROM IBM (OR ITS SUPPLIERS OR LICENSORS), OR ALTERING THE TERMS AND CONDITIONS OF ANY AGREEMENT OR LICENSE GOVERNING THE USE OF IBM PRODUCTS OR SOFTWARE.

© Copyright International Business Machines Corporation 2013. All rights reserved.