6 Distributed Agent Controller

The Distributed Agent Controller (also known as `opnet_appcapture`) is a command-line program that enables you to capture traffic without using Capture Manager. `opnet_appcapture` spawns a proxy process that manages the capture operation by sending commands (start capture, finish capture, download files) to all active agents.

Command-line captures offer the following advantages:

- You can create and run batch scripts that execute synthetic transactions and the capture processes simultaneously. Capturing a transaction by script (rather than manually) ensures that a packet trace records the exact start and stop times of the application.
- A batch script runs exactly the same way each time it is run, so you can capture multiple runs of the same transaction and eliminate user-related variations as a possible cause of the differences between different runs.
- Because the Distributed Agent Controller runs independently of AppTransaction Xpert, you can run and capture transactions from other computers in addition to the local computer running AppTransaction Xpert.

For more information, see:

- Licensing and Installation Requirements
- Installing and Configuring an Agent Controller
- Capturing a Transaction Using the Agent Controller
- Scheduling Application Transactions
- Capturing AppInternals Xpert Statistics Using Distributed Agent Controllers
- Troubleshooting Command-Line Captures

Licensing and Installation Requirements

To capture transactions using the Agent Controller, you must have a Distributed Agent Controller (DAC) license installed on a host that can communicate with the DAC computer.

If the Distributed Agent Controller and the AppTransaction Xpert software are installed on the same computer, or the DAC computer can communicate with a remote host that has AppTransaction Xpert installed, run AppTransaction Xpert and add the DAC license on the AppTransaction Xpert computer.

Note—For the Distributed Agent Controller to obtain the DAC license, the AppTransaction Xpert preference license_server_standalone must be set to FALSE on the computer where the DAC license is installed.

If the Distributed Agent Controller computer cannot communicate with any host that has AppTransaction Xpert installed, do the following:

1) Run the AppTransaction Xpert software installer on the DAC computer.

   Note—When the installer prompts for the type of licensing, select the “Standalone” option.

2) Run the AppTransaction Xpert software and add the DAC license on the Distributed Agent Controller host.

3) Uninstall the AppTransaction Xpert software from the Distributed Agent Controller host (*optional*).

Related Topics

- *Distributed Agent Controller*

Installing and Configuring an Agent Controller

Procedure 6-1 Installing and Configuring the Agent Controller

1 Run the Distributed Agent Controller installer.

To download the latest installer, go to https://support.riverbed.com.

2 Create one or more agent lists for the controller:

2.1 Create an agent list with all the agent settings that you want to use with the Distributed Agent Controller.

For more information, see Creating a Capture Agent/Appliance List.

2.2 Copy the agent list (*.agents) file from the AppTransaction Xpert computer to the Distributed Agent Controller directory.

3 Copy all packet filter (*.ace.f) files used by the listed agents to the configuration directory.

For more information, see Capture Manager.

End of Procedure 6-1

Related Topics

- *Distributed Agent Controller*

Capturing a Transaction Using the Agent Controller

Although you can capture traffic manually (by entering commands at a command line), the best practice is to create a script or batch file that runs commands in the order shown in the following figure.

Figure 6-1 Batch File for Capturing a Transaction: Example

```
# STEP 1: Start the capture
opnet_appcapture -start -task_name task_1 -agent_list my_agent_list.agents

# STEP 2: Run the transaction
my_synthetic_transaction.exe -arg_1 val_1 -arg_2 val_2

# STEP 3: When the transaction ends, finish the capture
opnet_appcapture -finish -task_name task_1

# STEP 4 (optional): If the packet traces were not downloaded automatically,
# download them to the Distributed Agent Controller computer
opnet_appcapture -download_task_console -task_name task_1 -move_files
```

Start a Capture Task

To start a capture process, enter the following command:

```
opnet_appcapture -start -task_name <taskname> -agent_list
<agent_list_name>
```

Table 6-1 Starting a Command-Line Capture: Arguments

Argument	Description
-start	Start the application capture operation
-task_name	Name of the capture task. opnet_appcapture stores the packet traces created during a task in the following directory: <*trace_file_dir*>/capture_files/<*task_name*>/<*console_location*> <*trace_file_dir*> is specified by the “trace_file_dir” preference in the `opnet_appcapture_settings.ef` file.
-agent_list	The name of the agent list file, which specifies the agents that will capture the transaction. This argument has the following requirements: • The file name must include the “.agents” suffix. • This file must be located in the directory specified by the “agent_list_path” preference in the `opnet_appcapture_settings.ef` file (by default, this directory is <*DAC_dir*>/configs). For more information about agent list files, see Creating a Capture Agent/Appliance List.

Finish a Capture Task

To finish a capture process, enter the following command:

```
opnet_appcapture –finish –task_name <taskname>
```

Table 6-2 Finishing a Command-Line Capture: Arguments

Argument	Description
-finish	Finish the capture task
-task_name	Name of the capture task to finish. If the agent list does not specify that the agent store the packet trace locally, the file is downloaded and saved in the following directory on the Distributed Agent Controller computer: <*trace_file_dir*>/capture_files/<*task_name*>/<*console_location*> <*trace_file_dir*> is specified by the “trace_file_dir” preference in the `opnet_appcapture_settings.ef` file. For more information about packet trace directories, see Managing Packet Traces.

Abort a Capture Task

To abort a capture process, enter the following command:

```
opnet_appcapture –abort –task_name <taskname>
```

Download Remote Files For a Capture Task

To download packet traces for a specific task, enter the following command:

```
opnet_appcapture -download_task_console -task_name <taskname> [-console_location
<location>] -agent_list <agent_list_name> [-move_files]
```

For each agent in the agent list, the DAC downloads the packet traces that correspond to the specified capture task (`task_name`) and the host that launched that task (`console_location`).

Table 6-3 Downloading Remote Files for a Capture Task: Arguments

Argument	Description
-download_task_console	Download all remote packet traces from the directory specified by the task name and console location.
-task_name	Name of the capture task that generated the files you want to download.
-console_location	Console name of the computer that started the capture. Use this argument if you want to download files for a capture task that was launched by a remote computer. In this case, you need to include the `console_location` argument so that the Distributed Agent Controller can download the correct files. If this argument is not included, the DAC assumes that the local computer launched the task, and downloads files of name <*`task_name`*> that were launched by the local computer.
-agent_list	The name of the agent list file, which specifies the agents that store the files you want to download. This argument has the following requirements: • The file name must include the ".agents" suffix. • This file must be located in the directory specified by the "agent_list_path" preference in the `opnet_appcapture_settings.ef` file (by default, this directory is <*DAC_dir*>/configs). For more information about agent list files, see Creating a Capture Agent/Appliance List.
-move_files	If this argument is included, the remote agents delete their copies of the downloaded packet traces.

For more information about packet traces, see Managing Packet Traces.

Download All Packet Traces from One or More Remote Agents

To download all packet traces residing on a remote agent, enter the following command:

```
opnet_appcapture -download_all -agent_list
<agent_list_name>.agents -move_files
```

Table 6-4 Downloading All Packet Traces: Arguments

Argument	Description
-download_all	Download all packet traces from all agents in the agent list
-agent_list	The name of the agent list file, which specifies the agents that will capture the transaction. This argument has the following requirements: • The file name must include the “.agents” suffix. • This file must be located in the directory specified by the “agent_list_path” preference in the `opnet_appcapture_settings.ef` file (by default, this directory is <*DAC_dir*>/configs). For more information about agent list files, see Creating a Capture Agent/Appliance List.
-move_files	If this argument is included, the remote agents delete their local copies of the downloaded packet traces.

Related Topics

- *Distributed Agent Controller*

Scheduling Application Transactions

Windows and Linux platforms support task scheduling, in which the operating system launches an application at scheduled times.

You can schedule tasks in Windows by clicking the “Scheduled Tasks” icon in the Windows Control Panel (choose Settings > Control Panel from the Windows Start menu). For more information, see the Windows documentation.

You can schedule tasks in Linux using the “cron” command-line utility. For more information, enter “man cron” from a command line.

To capture for a specific length of time, create two batch files: one file starts the capture, the other stops it. Then create two scheduled tasks. For example, you could create start.bat and stop.bat files; then you can schedule start.bat to run at 10pm, and stop.bat to run at 11pm, as shown in the following examples.

```
#start.bat file (example):
opnet_appcapture -start -task_name nightly_task -agent_list my_agent_list.agents
```

```
#stop.bat (example):
opnet_appcapture -finish -task_name nightly_task
opnet_appcapture -download_task_console -task_name nightly_task -move_files
```

Related Topics

- *Distributed Agent Controller*

Capturing AppInternals Xpert Statistics Using Distributed Agent Controllers

Use Distributed Agent Controllers to collect performance statistics directly from AppInternals Xpert as part of the traffic-capture process.

In AppInternals Xpert, a *correlation view* specifies a customized set of statistics for which you want to display data. When a capture operation finishes, the Distributed Agent Controller can retrieve data from AppInternals Xpert based on the capture time window and a correlation view.

Procedure 6-2 Collecting AppInternals Xpert Data Using a Distributed Agent Controller

1 Create a capture-agents list that includes information about the AppInternals Xpert server and correlation view:

1.1 Open Capture Manager and click the On-Demand Capture tab.

1.2 Load, create, or edit the capture-agent list that you will use for the Distributed Agent Controller.

1.3 Select the "Collect performance statistics from AppInternals Xpert" option and click Configure.

➥ The Configure AppInternals Xpert Collection dialog box appears.

1.4 Specify the AppInternals Xpert hostname, the username and password (needed for AppTransaction Xpert to log on to the server), and the correlation view for the data you want to collect. Then click OK to return to Capture Manager.

1.5 Click Save Agent List and specify a file name.

➥ The resulting file (<*filename*>.agents) includes the specified AppInternals Xpert information.

2 Verify that AppInternals Xpert is running and that the correlation view is active.

3 Run the capture operation using the Distributed Agent Controller and the agents list that includes the AppInternals Xpert data.

➥ When the capture operation finishes, AppTransaction Xpert does the following:

- Downloads performance data from the AppInternals Xpert server, based on the time window of the capture and the selected correlation view.
- Stores the AppInternals Xpert data as a .pan file in the same directory as the packet traces generated during that operation.

4 To view the AppInternals Xpert data in AppTransaction Xpert, do the following:

- Import the packet traces into AppTransaction Xpert, as described in Creating a Transaction Analyzer Model.
- To import the AppInternals Xpert data into an existing Transaction Analyzer model, do Importing Performance Statistics.

End of Procedure 6-2

For more information about correlation views, see the AppInternals Xpert documentation.

Related Topics

- *Distributed Agent Controller*

Troubleshooting Command-Line Captures

When `opnet_appcapture` completes a capture successfully, the agents store any resulting packet traces in the locations indicated in Table 4-9 Packet Trace Storage Locations.

If `opnet_appcapture` encounters an error during a capture, it generates a log message to the file “<*DAC_dir*>/err_log.” If you experience problems during a capture, review this log.

If you still cannot resolve the problem, try repeating the capture while in verbose mode. To do this, open the opnet_appcapture_settings.ef file and set the `-verbose_log` setting to `TRUE` (or add this setting if it is not specified in the file). When you do a capture with this option enabled, opnet_appcapture generates status messages to the log files “opnet_appcapture.log” and “opnet_appcapture_proxy.log”, which are located in your <*DAC_dir*> directory. When the capture finishes, check these log files for messages.

Related Topics

- *Distributed Agent Controller*