Application Reporting Tool

Table of Contents

Application Reporting Tool

Help version 1.020

CONTENTS

Overview

The Application Reporting Tool is a technical support utility, available to Windows Administrators on the installation machine, that automates the process of gathering and archiving data for troubleshooting or identifying system issues. With the exception of identifying hardware and gathering general operating system information, the Application Reporting Tool only collects file and registry information that directly pertains to related applications. After report generation, the entire report archive is stored in a user-specified location and may be reviewed before sending to technical support.

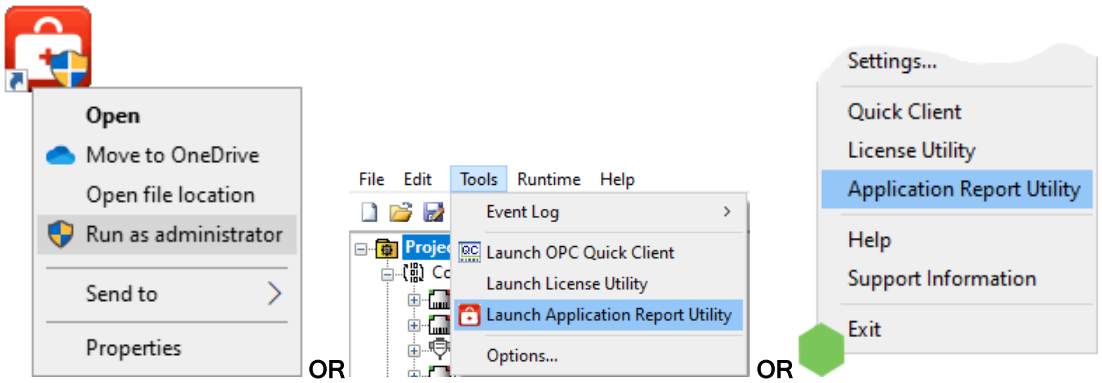
*For more information regarding what data is collected as part of an Application Report archive, see **Information Included in an Application Report**.*

See Also:
Generating an Application Report

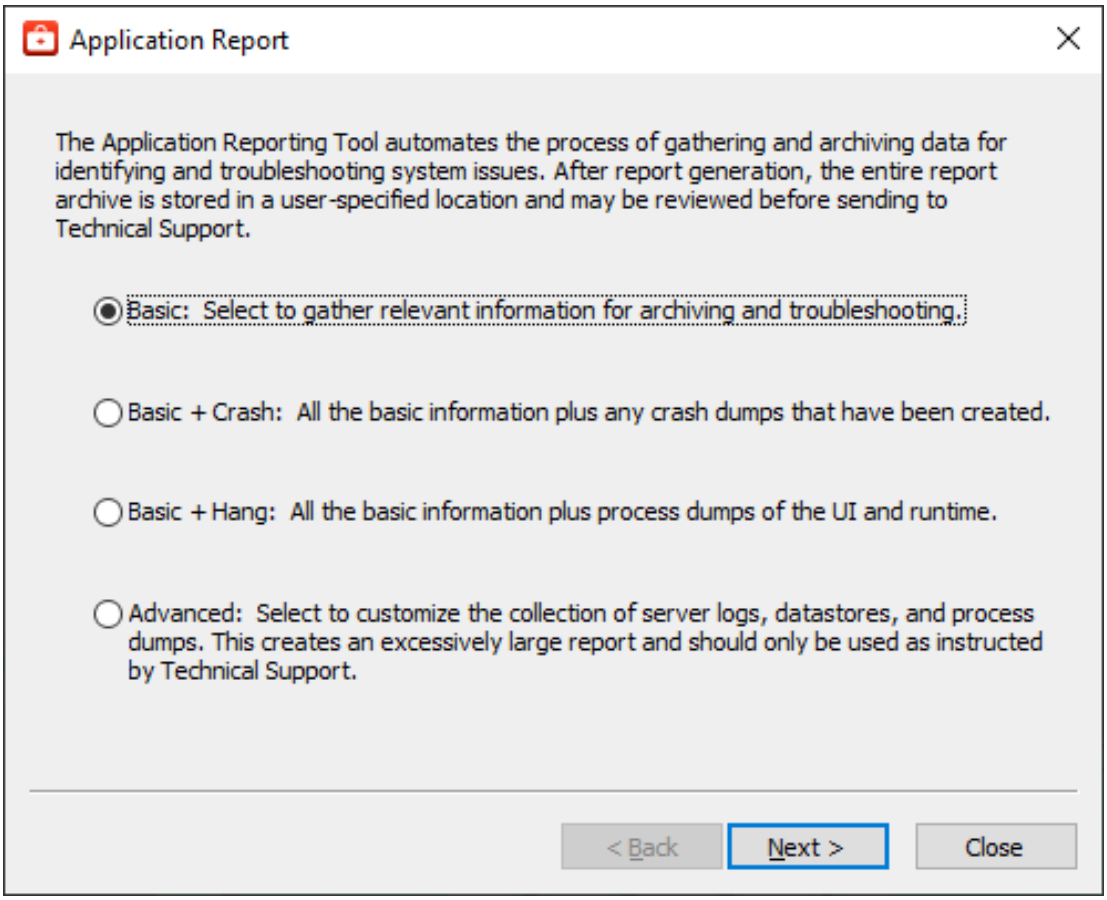
Generating an Application Report

To create a report with the Application Reporting Tool.

1. Launch the **Application Report** tool.
   As the Administrator through the desktop icon OR through the server **Tools** menu OR through the **Administration** menu.

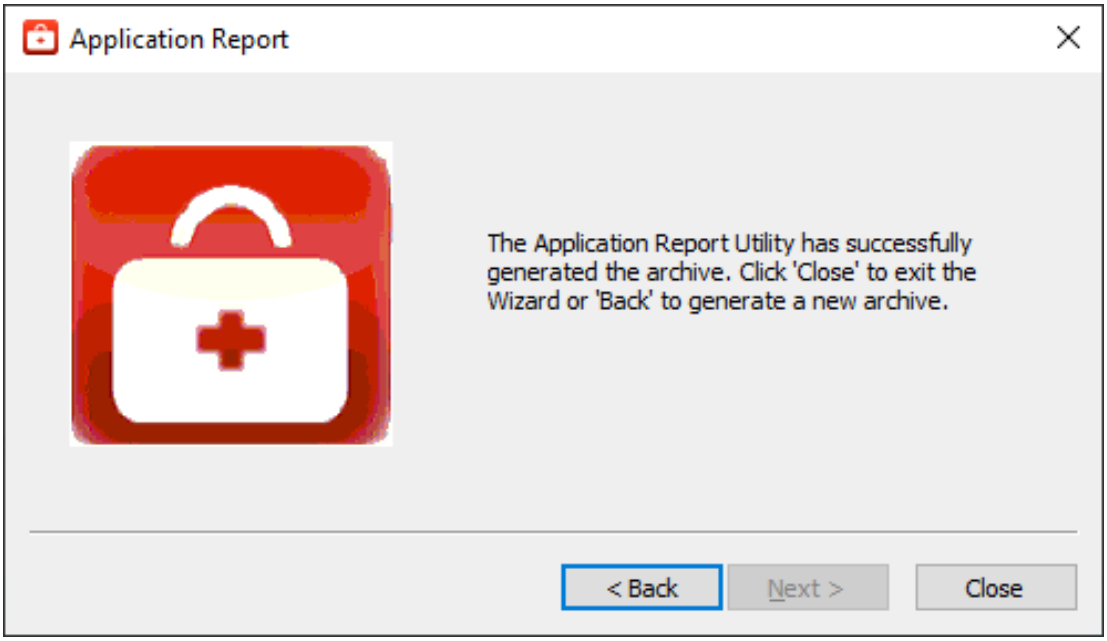


2. Select **Basic**, **Basic + Crash**, or **Basic + Hang**. Basic gathers all the normal troubleshooting information. If the application is experiencing crashes, use **Basic + Crash**. If the application is in a deadlocked state, use **Basic + Hang**. (*(To use Advanced, see steps below.)*



3. Click **Next >**.
4. Review the information to be gathered. Click **Next >**.
5. Choose the location for the file to be saved and click **OK**.

6. The Application Reporting Tool collects the files, generates the archive, and displays a notification message on completion.

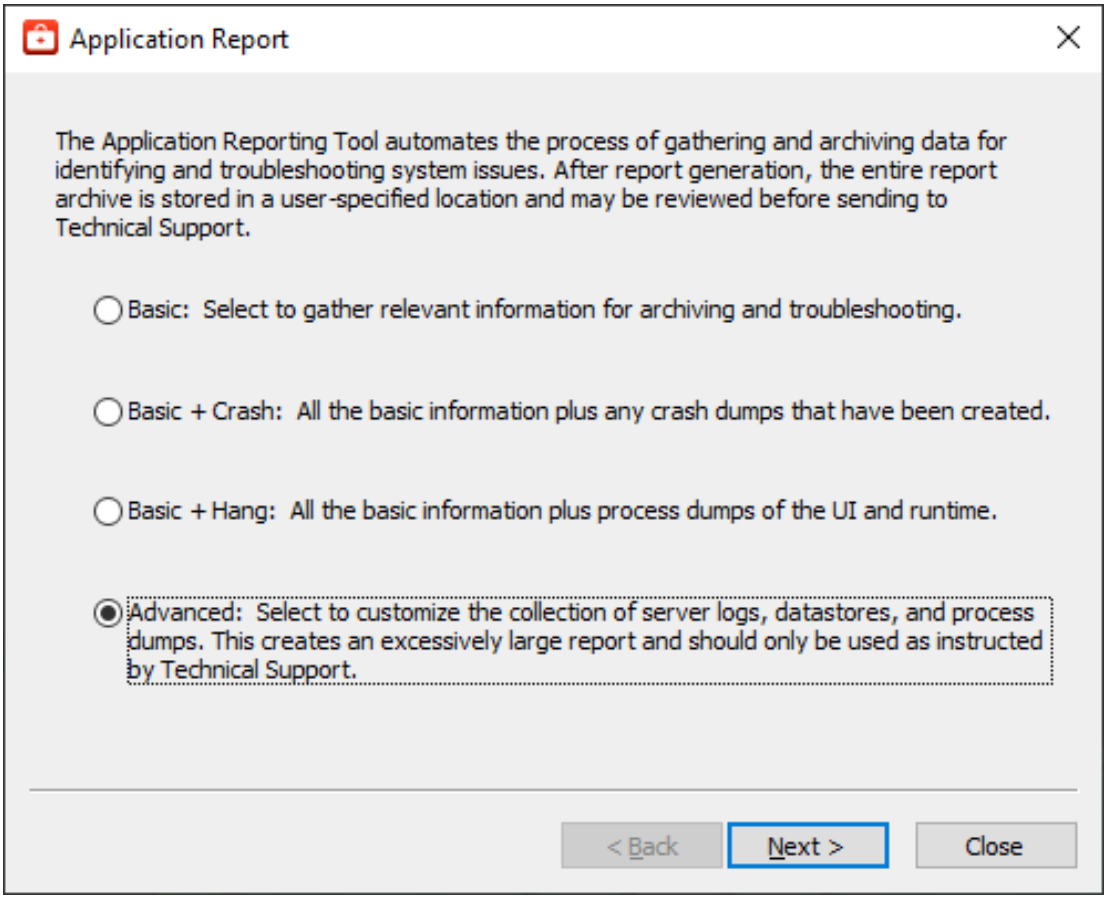


7. Click **Close**.
8. Browse to the archive in the output folder. Output archives are compressed using the ZIP format and follow the naming convention (UTC Time):

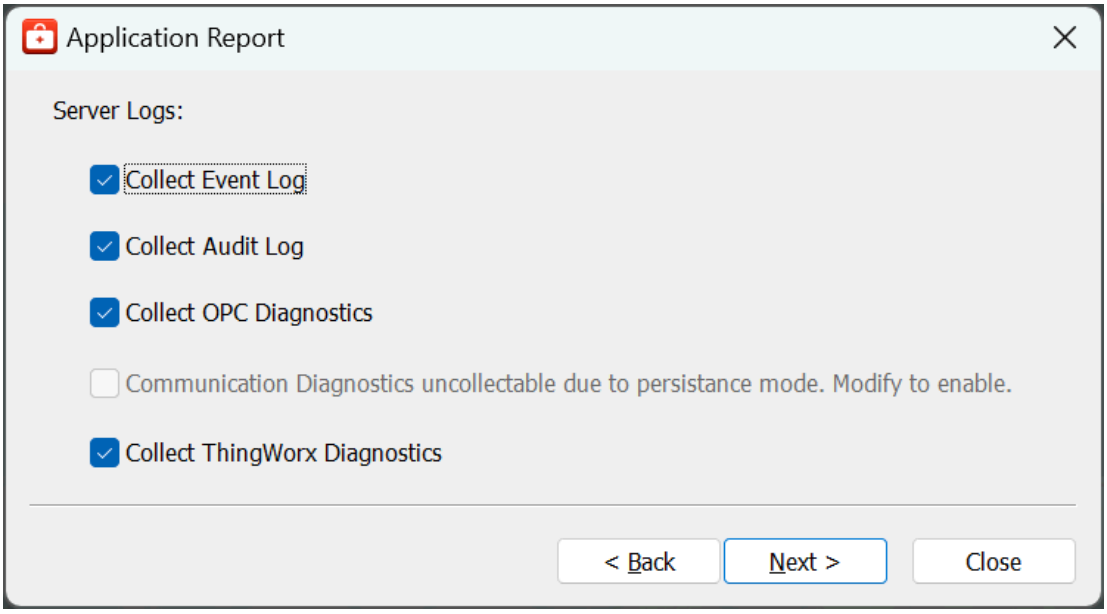
   ARU_YEAR-MONTH-DAY_HOUR-MIN-SEC.zip
9. Send the file to Technical Support or extract the archive and browse the directory for the files of interest.

Advanced

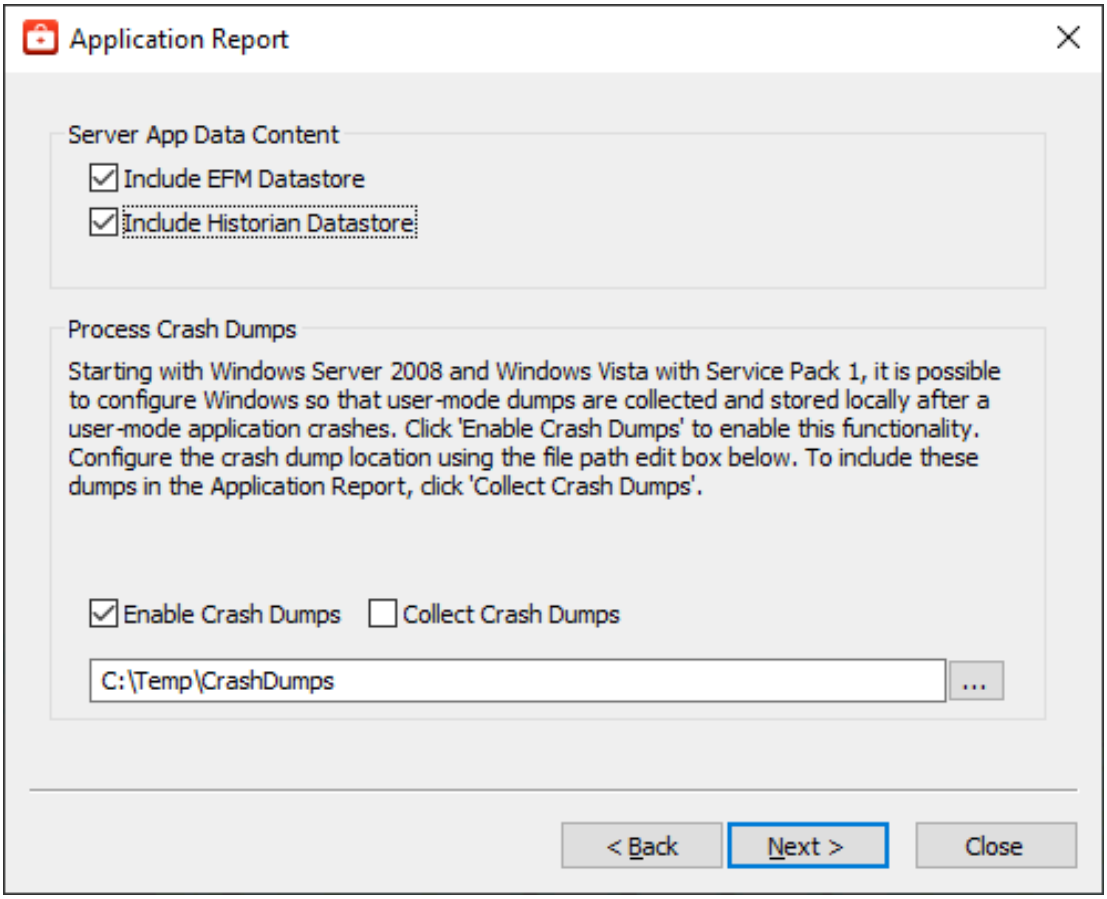
1. Select **Advanced** to control how much detail to include and click **Next**.



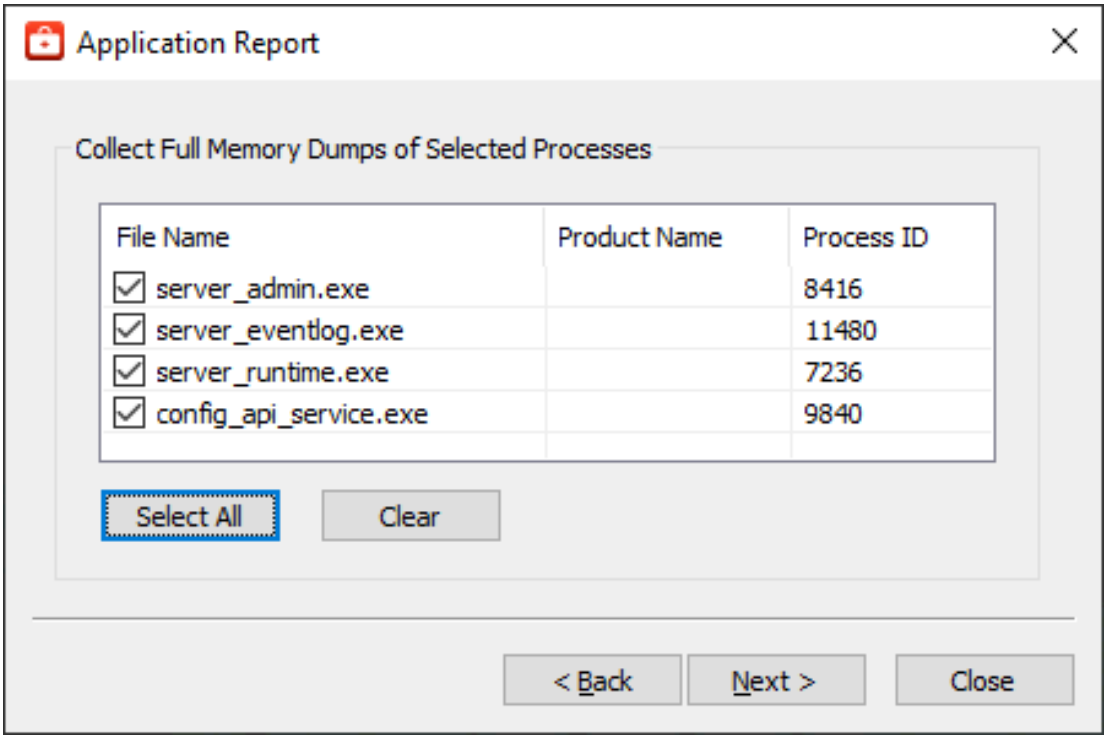
2. Choose the event logs and diagnostics to collect, as described in Collecting Event Logs and click **Next**.



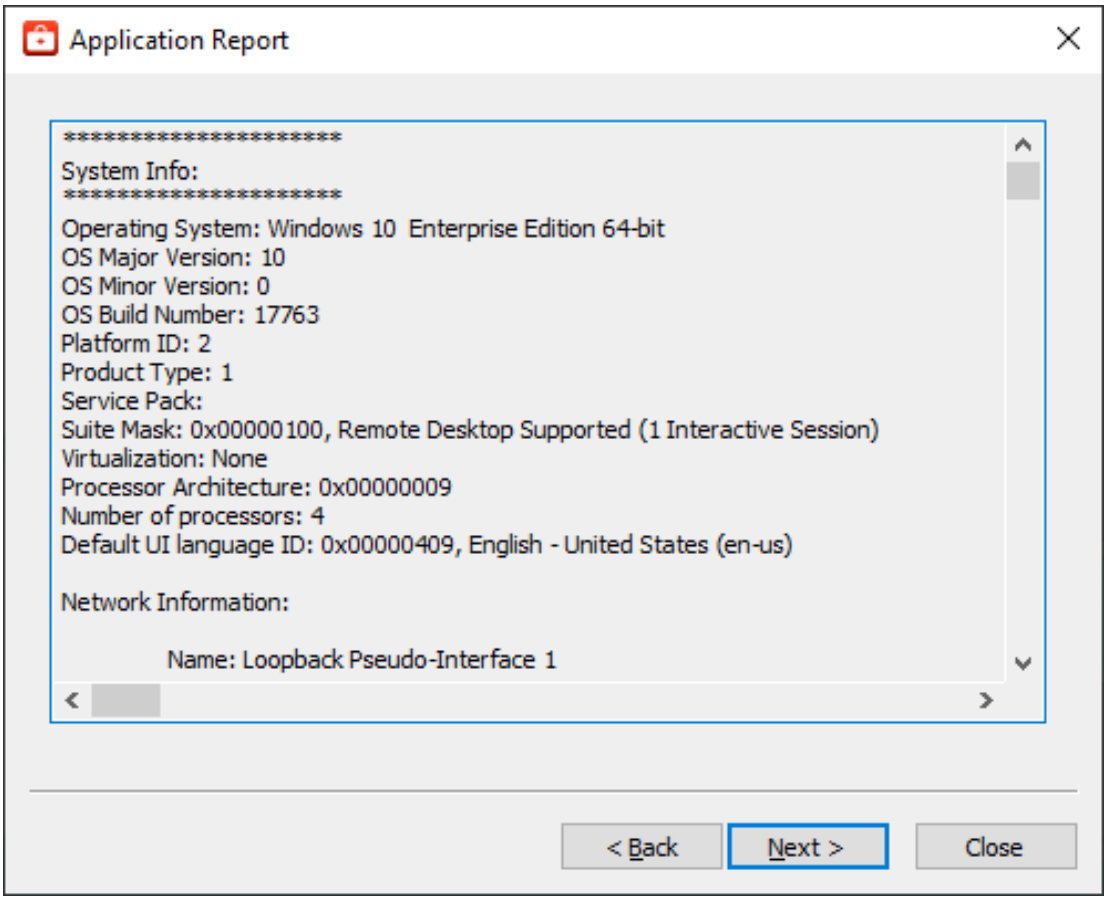
3. Select content to include as described in Server Data and Crash Dumps.



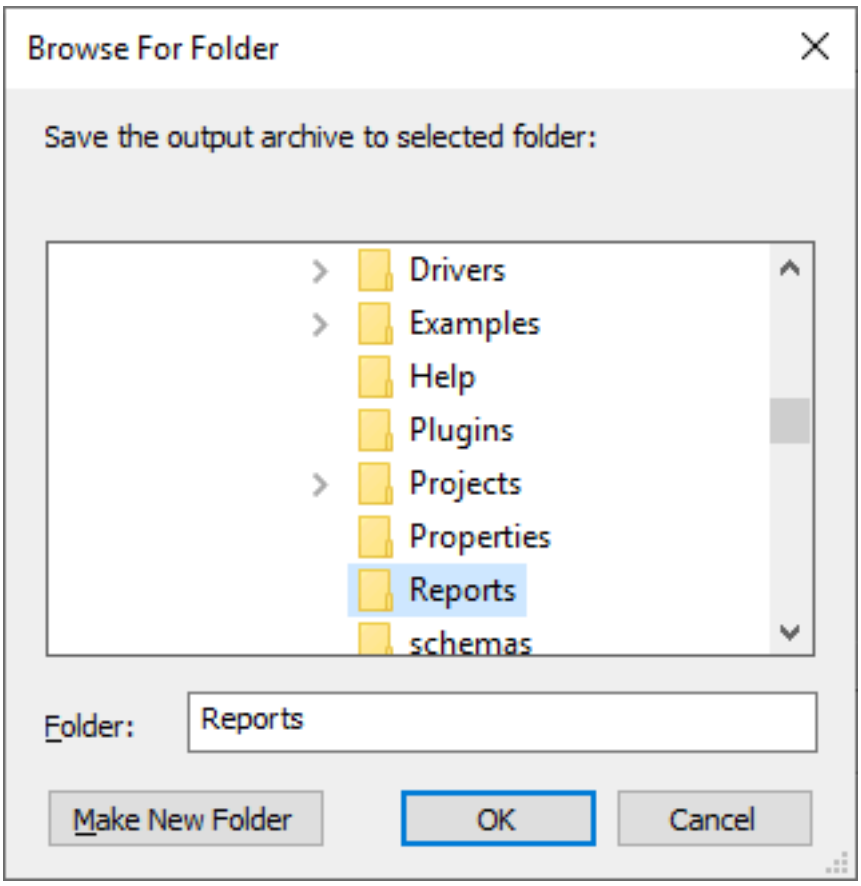
4. Click **Next**.
5. Select the processes for which to generate on-demand process memory dumps and click **Next**.
   For more information on process memory dumping, including when a memory dump is necessary, refer to Server Data and Crash Dumps.



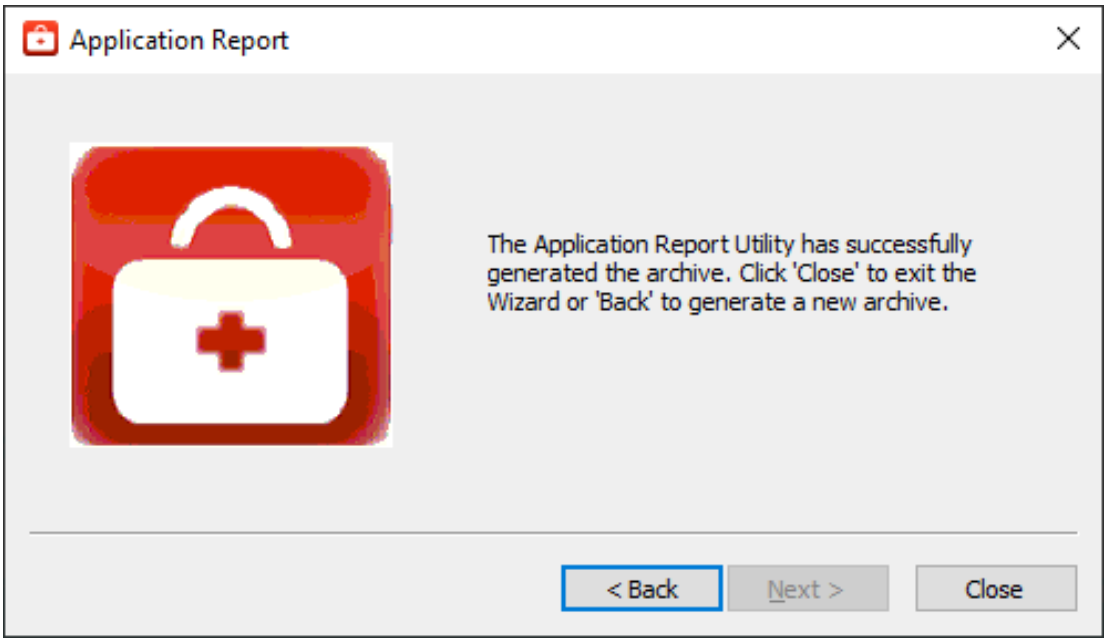
6. The Application Reporting Tool generates a preview of the collected data for review. Verify the settings and click **Next** (or click **Back** to make changes before generating).
   For details regarding the information collected, see Information Included in an Application Report.



7. Browse to and select the folder in which to save the report archive.



8. Click **OK**.
9. The Application Reporting Tool collects the files, generates the archive, and displays a notification message on completion.



10. Click **Close**.

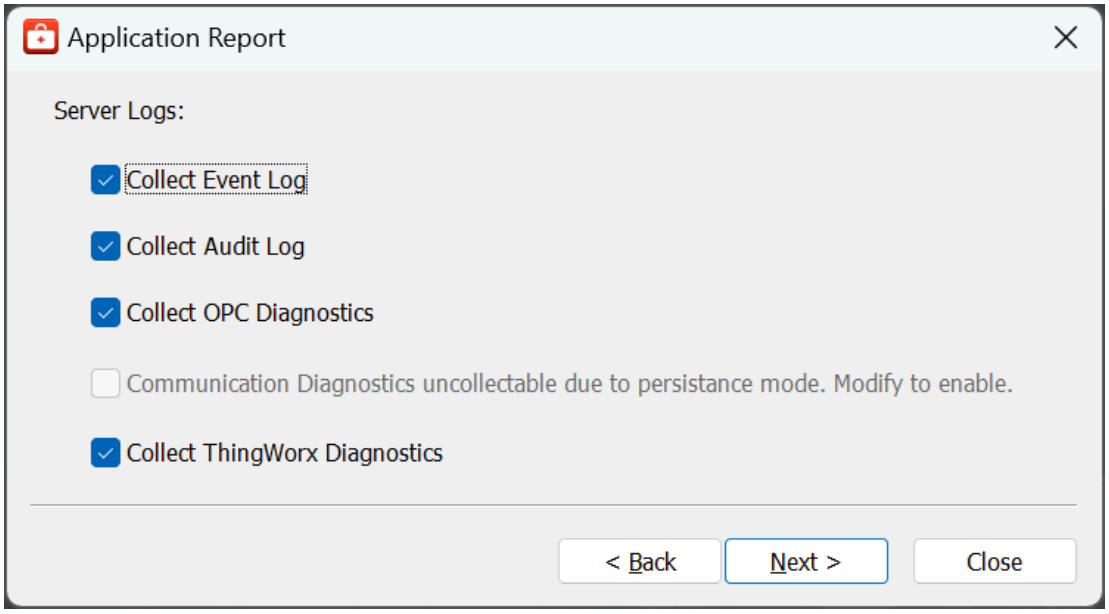
11. Browse to the archive in the output folder. Output archives are compressed using the ZIP format and follow the naming convention (UTC Time):

    ARU_YEAR-MONTH-DAY_HOUR-MIN-SEC.zip

12. Send the file to Technical Support or extract the archive and browse the directory for the files of interest.

Collecting Event Logs

An application-specific, user-configurable event logging service is included in most products. The technical support team may request the logs generated by the service to better understand the error and any relevant diagnostic information provided by the product.



The log collection interface is divided into two sections, allowing collection of server logs and LinkMaster logs. If either product is not installed, the section is disabled.

Server Logs

Five types of server logs are collectable:

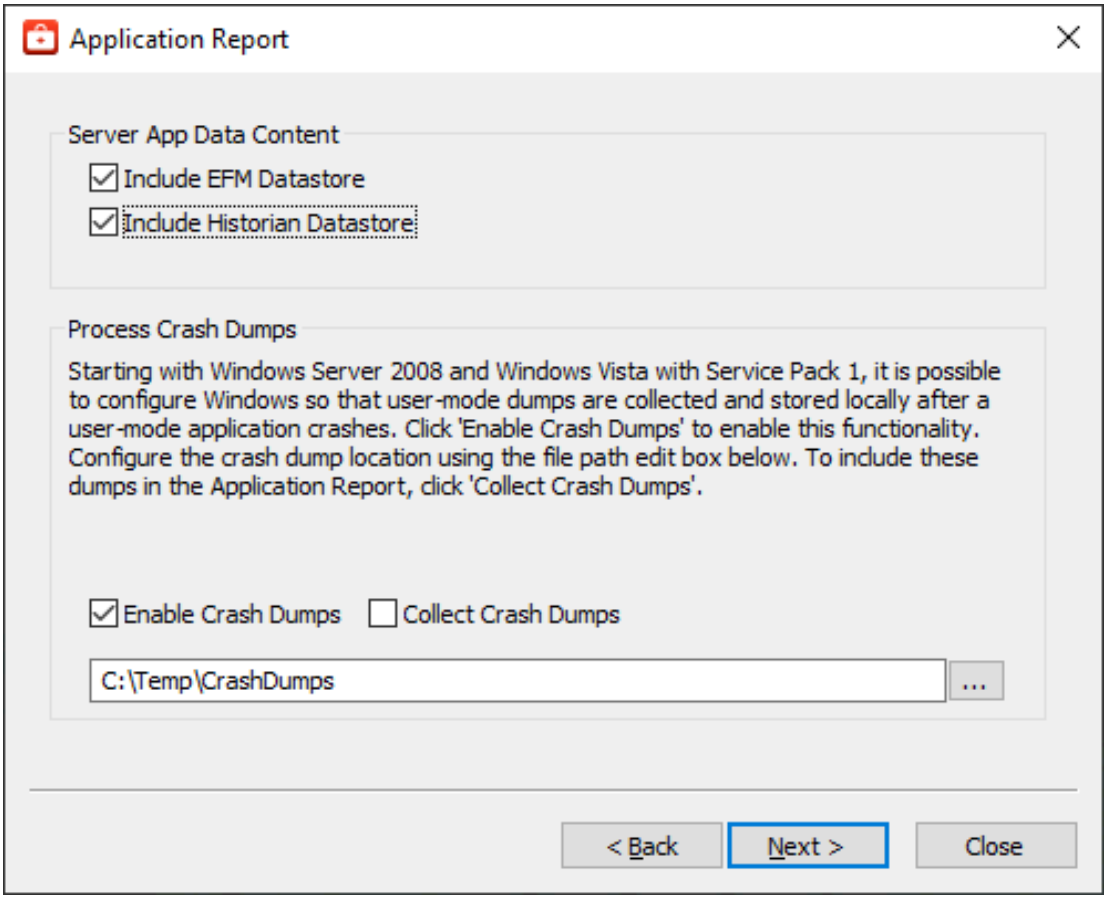
- **Collect Event Logs** Records noteworthy occurrences at the server level.
- **Collect Audit Log** Records changes to the server project.
- **Collect OPC Diagnostics** Records OPC events occurring between an OPC client and the server.
- **Collect Communication Diagnostic Records** record messages and events occurring between a driver and a device.
- **Collect ThingWorx Diagnostics**: Records native interface events and messages between the server, the CSDK, and the ThingWorx Platform.

LinkMaster Logs

- **Collect Event Logs** Records noteworthy occurrences at the server level.

Server Data and Crash Dumps

There are several choices that affect the size and completeness of the report.



Server App Data Content

Many Windows applications leverage the Application Data directory as an area for storage of temporary or long-term files that do not require direct user interaction. The Application Reporting Tool collects this directory to provide technical support staff with a better understanding of the state of applications on the system. Plug-ins installed as part of the server product also use this space for storage. This information is not always needed by technical support, but can be included when necessary.

Include EFM Datastore: Server's EFM Suite stores its historical EFM (Electronic Flow Measurement) data within the Application Data directory. Selecting this option allows the collection of EFM content when collecting the Application Data directory. If no EFM content exists within Application Data, this selection has no effect.

Include Historian Datastore: The server's local "historian" plug-in may store its database in any location, including the Application Data directory. Selecting this option allows the collection of historian datastore files when collecting the Application Data directory. If no historian datastore exists within Application Data, this selection has no effect.

Process Crash Dumps

Windows Vista SP1/Server 2008 releases and higher provide the ability to generate process memory dumps automatically when a process crashes, providing valuable insight into the conditions leading to the crash. The Application Reporting Tool configures the system to collect ONLY those dumps related to this particular vendor software.

Enable Crash Dumps: Sets / disables a system-wide registry key, notifying Windows to generate a process memory dump any time a process crashes. Within the text field, a default path of C:\Temp\CrashDumps is provided and may be changed to any location at any time.

Collect Crash Dumps: Process memory dumps related to vendor products that are stored in the selected path are collected as part of the Application Report archive. Within the archive, process memory dumps appear in the /CrashDumps folder of related products.

Note: Collecting process memory dumps require administrative privileges. If the system or authorized user does not have adequate privileges, the utility requests temporary elevation of rights to administrator level.

Information in an Application Report

As part of the Application Reporting Tool, many different pieces of information are included. Below is a list of some of the information and files collected as part of archive generation.

System

- Information Compiled:
  - Hardware Details
  - Operating System Details
  - Active and Disconnected Network Interfaces
  - Installed .NET Frameworks
  - DCOM State and Permissions
  - OPC Enum Service Details
  - Registered OPC Servers (as seen by OPC Enum)
- Files Copied:
  - bootstrap.log
    - Error log generated during the failure of any Windows installer application
  - <AppData>\Vendor\Common
    - Vendor Hardware Keys
  - <AppData>\FLEXnet
    - Vendor Licensing
  - Windows System Event Log File
  - Windows Application Event Log File

General Product

- Information Compiled:
  - Installed Components
    - A list of .exe and .dll files stored in the install directory of each product
  - Xi Wrapper (Server Only)
  - Product Registry Entries
    - HKEY_CURRENT_USER\SOFTWARE\<Vendor>\<Product>\V5
    - HKEY_LOCAL_MACHINE\SOFTWARE\<Vendor>\<Product>\V5
    - HKEY_CLASSES_ROOT\AppID\<Product_CLSID>
    - HKEY_CLASSES_ROOT\CLSID\<Product_CLSID>
  - Product DCOM Configuration and Permissions
- Files Copied:
  - Trusted Storage Diagnostics
    - License details file generated by a product's "activation_client.exe"
  - Install Log
    - The log file generated by each product during installation and modification
  - (Optional) Log Files
    - Includes Event, Audit, OPC Diagnostics, Communication Diagnostics, and ThingWorx Native Interface logs *See "Collecting Event Logs" on page 10*
  - Application Data
    - Temporary and long-term storage for application specific files
    - Log files from the Program Files directory (RedundancyMaster Only)

Other

- (Optional) On-Demand Process Memory Dumps
  - *See **Server Data and Crash Dumps***

- (Optional) Process Memory Crash Dumps
  - *See Server Data and Crash Dumps*

Command Line Options

The Application Reporting Tool can be run from a command line. The following section provides the commands and options available.

Command : processlist, proclist, pl

Description: Display a list of running processes.

Arguments: none

Flag: none

Example: ./applicationreport.exe pl

Command : output, out, o

Description: Save the output of the Application Reporting Tool. This can take several minutes to complete. Results are stored in the output directory as a compressed .ZIP file.

Arguments: <File Path> : <Path to the output directory>

Flags: < PID> pid/processid/procid: Given process IDs, the Application Reporting Tool includes the full process dumps in the archive.

Example ./applicationreport.exe o "C:\ARU_Output" 8596

Command : dump, dumpall, all

Description: Saves all processes and includes them in the archive.

Arguments: <Any output command> <File Path> : <Path to the output directory>

Flag: none

Example: ./applicationreport.exe o "C:\ARU_Output" dump

Command : help , h

Description: Display the explanations for the commands.

Arguments: none

Flag: none

Example: ./applicationreport.exe help

Index

I

L

N

O

S

T

Z