Exchange Data Collector (ExDC) v4.0.2

*This is written for v4.0.2*

**Overview**

ExDC is used to collect a large amount of information about an Exchange environment with minimal effort by the admin. The data is initially collected into a series of text files that can then be assembled into Excel spreadsheets and a Word document on the data collection server or another workstation. The four output files are:

* ExDC\_DC – Spreadsheet of WMI and registry settings on domain controllers
* ExDC\_Exch – Spreadsheet of WMI and registry settings on Exchange servers
* ExDC\_ExOrg – Spreadsheet of Exchange organization cmdlet output
* ExDC\_ExchangeEnvironmentalReport – Document of the summary of Exchange environment

**Prerequisites**

The Data Collection server must meet the following requirements:

1. **Operating System :** The operating system required is based on the version(s) of the Exchange Server installed in the environment
2. **Regional Settings :** The workstation locale needs to be set to English-X where X is any of the regions supporting an English locale
3. **Minimum Resolution :** 1024 x 768
4. **Powershell :** Powershell version 2 or later is required
5. **Exchange Management Tools :** If Exchange Server 2010, Exchange 2013, or Exchange 2016 is installed in the environment, the corresponding Exchange tools must be installed. If multiple versions are installed in the environment, the latest version of the Exchange Management Tools must be installed on the server

The server or workstation being used to assemble the data into Office reports must meet the following requirements:

1. **Operating System :** 32-Bit or 64-Bit version of the Windows 8.1, Windows 10, Windows Server 2008 R2, Windows Server 2012, Windows Server 2012 R2, Windows Server 2016 (i.e. all Microsoft Operating Systems in the support lifecycle)
2. **Regional Settings :** The workstation locale needs to be set to English-X where X is any of the regions supporting an English locale
3. **Minimum Resolution :** 1024 x 768
4. **Powershell :** Powershell version 2 or later is required
5. **Excel :** Excel 2010 or later must be installed

**ExDC Usage**

The ExDC supports the following optional parameters:

**.PARAMETER JobCount\_ExOrg**

**Max number of jobs for Exchange cmdlet functions (Default = 10)  
Caution: The OOB throttling policy sets PowershellMaxConcurrency at 18 sessions per user per server. Modifying this value without increasing the throttling policy can cause ExOrg jobs to immediately fail.**

**.PARAMETER JobCount\_WMI**

**Max number of jobs for non-Exchange cmdlet functions (Default = 25)**

**.PARAMETER JobPolling\_ExOrg**

**Polling interval for job completion for Exchange cmdlet functions (Default = 5 sec)**

**.PARAMETER JobPolling\_WMI**

**Polling interval for job completion for non-Exchange functions (Default = 5 sec)**

**.PARAMETER Timeout\_ExOrg\_Job**

**Job timeout for Exchange functions (Default = 3600 sec)  
The default value is 3600 seconds but should be adjusted for organizations with a large number of mailboxes or servers over slow connections.**

**.PARAMETER Timeout\_WMI\_Job**

**Job timeout for non-Exchange functions (Default = 600 sec)  
If a job exceeds this value, it is terminated at the next interval specified in JobPolling\_WMI. The default value is 600 seconds but should be adjusted for organizations with servers over slow connections.**

**.PARAMETER ServerForPSSession**

**Exchange server used for Powershell sessions**

**.PARAMETER INI\_Server**

**Specify INI file for Server Tests configuration**

**.PARAMETER INI\_Cluster**

**Specify INI file for Cluster Tests configuration**

**.PARAMETER INI\_ExOrg**

**Specify INI file for ExOrg Tests configuration**

**.PARAMETER NoEMS**

**Use this switch to launch the tool in Powershell (No Exchange cmdlets)**

**.PARAMETER NoGUI**

**Use this switch with template files specified automatically start data collection.  
Any additional files required (mailbox.txt, dc.txt, and/or exchange.txt) must be present. The -ServerForPSSession parameter must also be specified. This is useful for scheduling the data collection using Task Scheduler.**

**.PARAMETER NoGUIOutputFolder**

**This switch is only used when the -NoGUI switch is used.  
When specified, the output folder will be renamed to this value after data collection completes. If not specified, the output folder will be rename to "output-x" where x is the Get-Time tick value**.

**Installing ExDC on the Data Collection Server**

1. The most current version of ExDC can be found at: https://gallery.technet.microsoft.com/Exchange-Data-Collector-ed48c3db   
   Development builds are located in branches on the GitHub repository:  
   <https://github.com/stemy-msft/exchange-data-collector/>
2. Check the properties of the zip (Figure 1) and insure that the file is not blocked. If the file is blocked, click on the “Unblock” button before proceeding to the next step. If this is not done, the execution of the scripts may be unexpectedly blocked.
3. Unzip the file to a working path with sufficient disk space. The actual space requirement is based on the amount of data collected but should be less than 100MB in most environments.

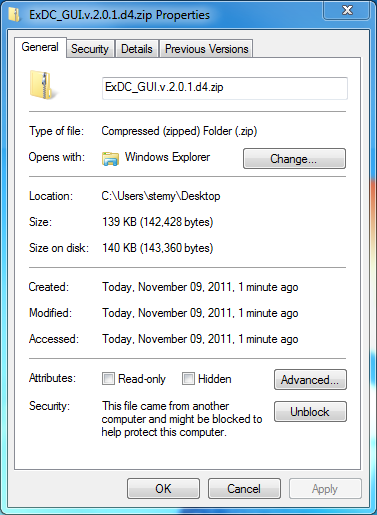


Figure 1: Blocked File

**ExDC Layout**

### 

The ExDC GUI consists of 4 main sections – the menu, the main tab group, the activity area, and the status bar.

### The Menu

The menu section has three main options – File, Toggle, and Help.   
**File -> Load all Targets from File**

This will quickly load Domain Controllers from DC.txt, Exchange Servers from Exchange.txt, Exchange Nodes (if applicable) from ClusterNodes.txt, and Mailboxes (if applicable) from Mailbox.txt.

**File -> Package application log**

This selection will save all “ExDC” events from the application log in the last 24 hours to an ExDC\_Events\_<TimeStamp>.txt file. After the application events have been saved to the text file, it will create a zip file called ExDCPackagedLogs\_<TimeStamp>.zip containing all ExDC\_Events\*.txt files, all ExDC\_Step3\_Transcript.txt files, and all FailedPing\*.txt files.

**Toggle -> Check All Targets**

This will check all checkboxes on the Targets tab

**Toggle -> Uncheck All Targets**

This will uncheck all checkboxes on the Targets tab

**Toggle -> Check All Tests**

This will check all checkboxes on the Tests tab

**Toggle -> Uncheck All Tests**

This will check all checkboxes on the Tests tab

**Help -> Help**

This will open the web page version of this help file

**Help -> About**

This will display version and release date information on this build of ExDC

### The Main Tab Group

Immediately under the Menu, the Main Tab Group has four tabs – Targets, Templates, Tests, and Reporting

### The Activity Area

The Activity Area takes up most of the area of the ExDC GUI. When Targets is selected, the Activity Area will have tabs for Domain Controllers and Exchange Servers. Depending on the environment, an Exchange Nodes and/or Mailboxes tab may also be present. When Tests is selected, the Activity Area will have an Execute button and a tab for Server Functions. Depending on the Environment, a second tab – ExOrg Functions – may also be present. When Reporting is selected, the Activity Area will have an Execute button and checkboxes to select the reports to generate.

### The Status Bar

The Status Bar will indicate the status of the activity on Targets, Tests, or Reporting.

**Using ExDC to Collect Data**

### Scenario 1: Pure or mixed Exchange 2010/2013/2016

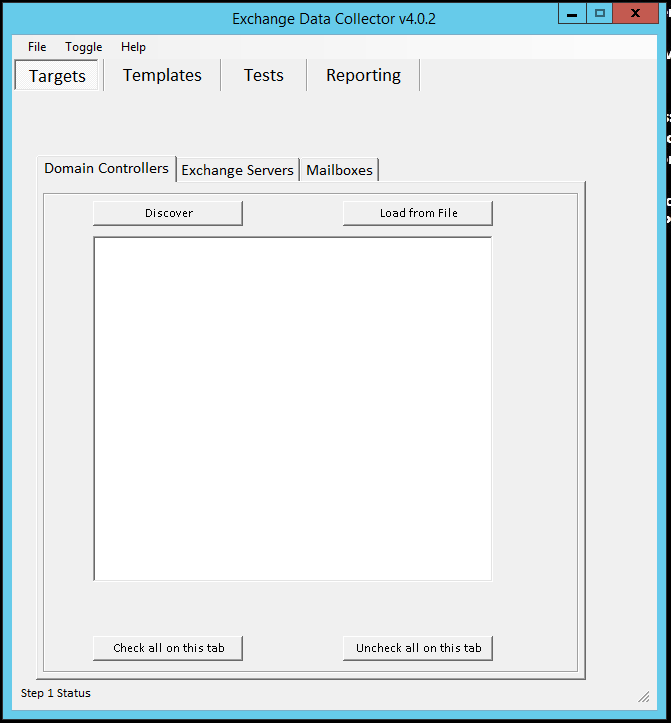
1. Open the latest version of the Exchange Management Shell
2. Set the execution policy to be unrestricted by typing:

Set-ExecutionPolicy Unrestricted

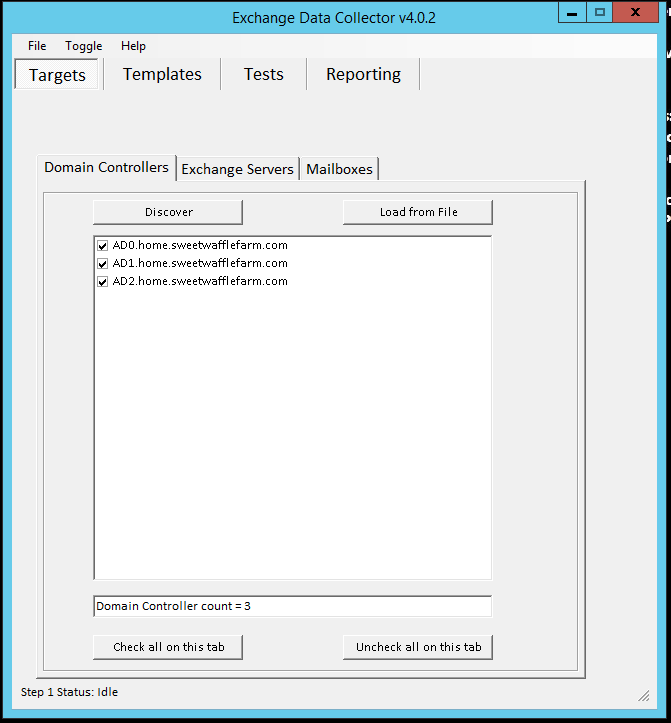
1. In the Powershell window, navigate to the working path of the ExDC files
2. Run ExDC with the following command:

.\ExDC.v4.ps1

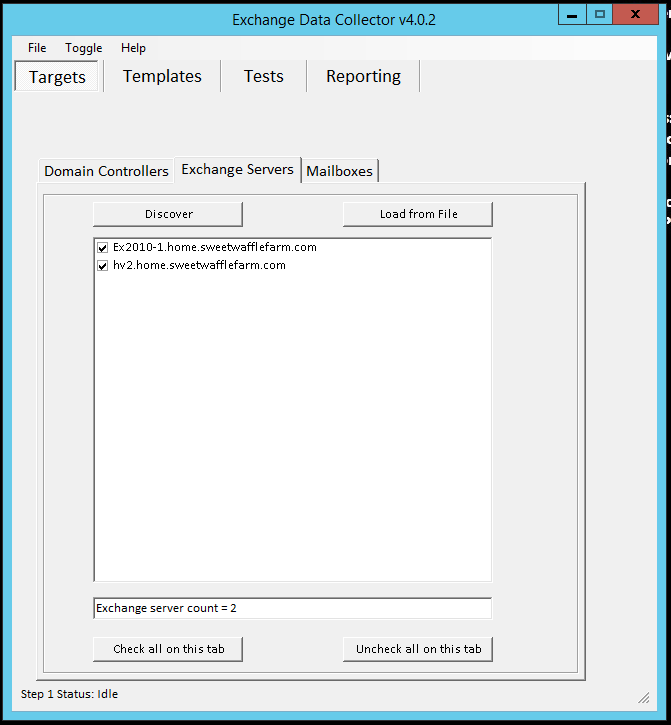
1. Once the form initializes, the three main tabs – Targets, Tests, and Reporting – will be visible at the top.



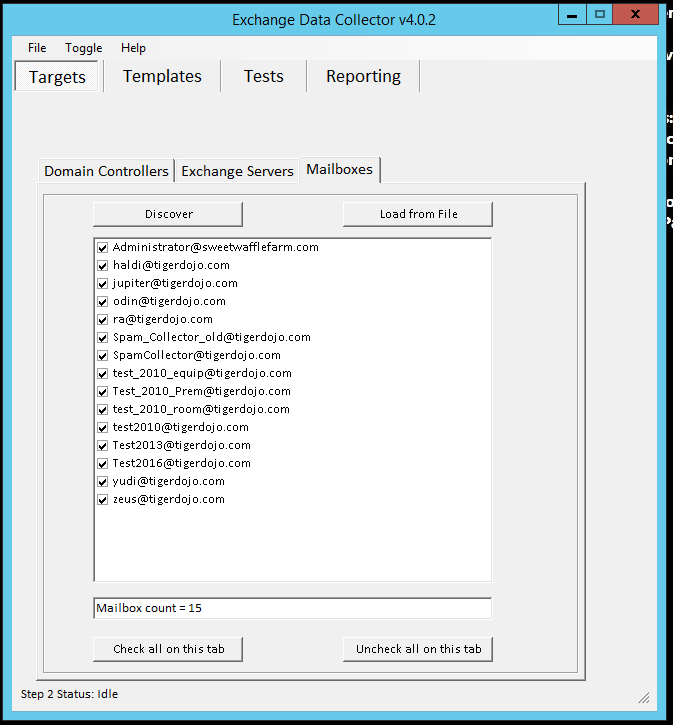
1. On the Domain Controllers tab, click the “Discover” button to discover all domain controllers in the forest. This will create a dc.txt file in the working path. If the dc.txt file has already been created, the “Load from File” button should be used to load the contents of the file into checked list box. Once discovery completes, undesired servers can be unchecked from the list.



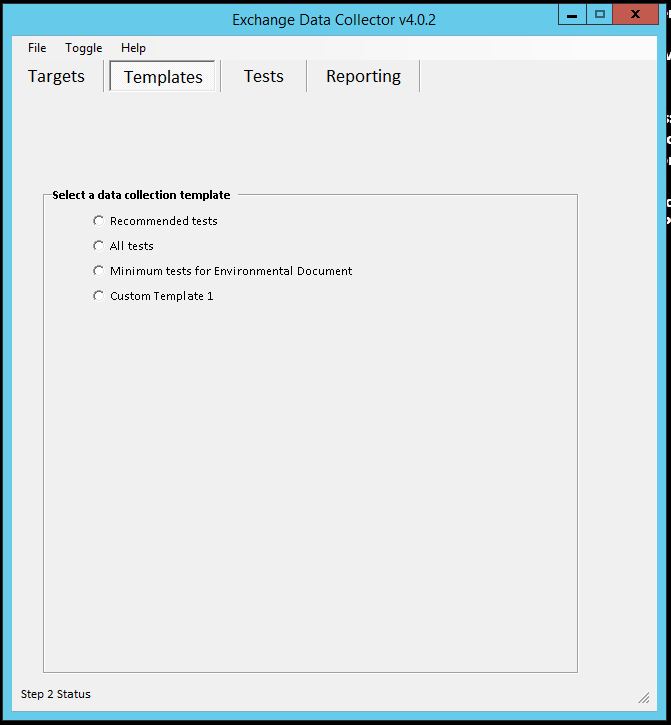
1. On the Exchange Servers tab, click the “Discover” button to discover all Exchange servers in the forest. This will create an exchange.txt file in the working path. If the exchange.txt file has already been created, the “Load from File” button should be used to load the contents of the file into checked list box. Once discovery completes, undesired servers can be unchecked from the list.



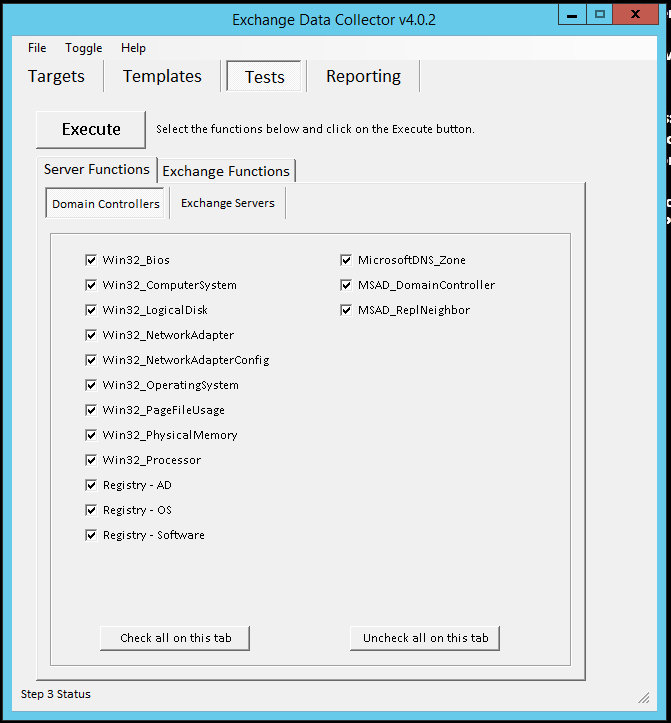
1. On the Mailboxes tab, click the “Discover” button to discover all mailboxes in the forest. This will create a mailbox.txt file in the working path. If the mailbox.txt file has already been created, the “Load from File” button should be used to load the contents of the file into checked list box. Once discovery completes, undesired mailboxes can be unchecked from the list.



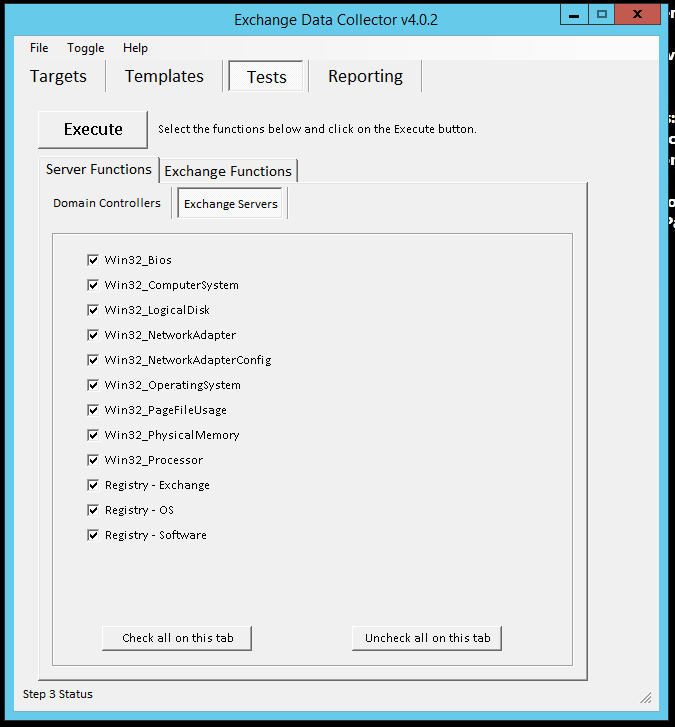
1. If desired, select the “Templates” tab at the top of the window. Selecting a template here will check/uncheck all tests specified in the selected template. The “Recommended tests” template will give a good overview of the environment but will not run some of the more time-consuming tests (like Get-MailboxFolderStatistics). The “Custom Template 1” allows admins to customize templates for a set of tests that they would frequently run by modifying the /Templates/Template\_Custom1\*.ini files.



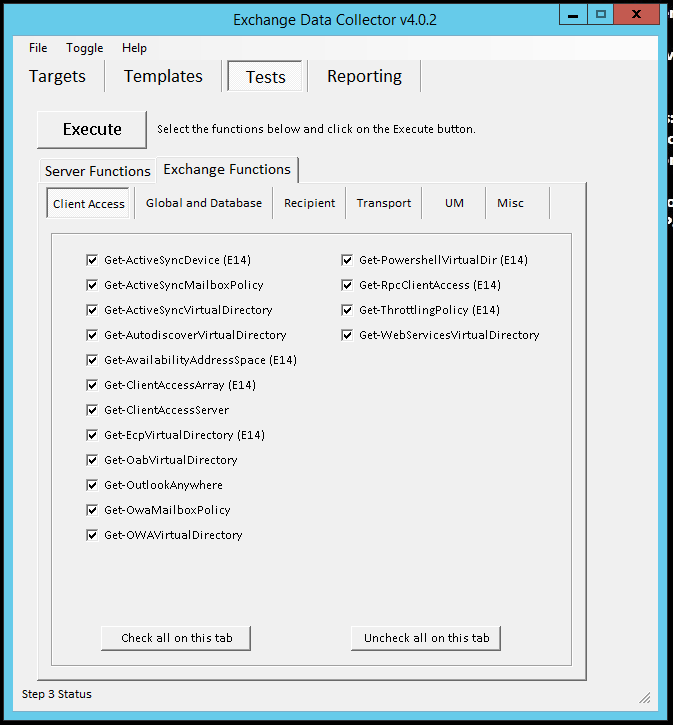
1. Select the “Tests” tab at the top of the window. The Tests are divided into Server Functions and Exchange Functions.
2. Under Server Functions, insure that the desired functions for Domain Controllers are selected.



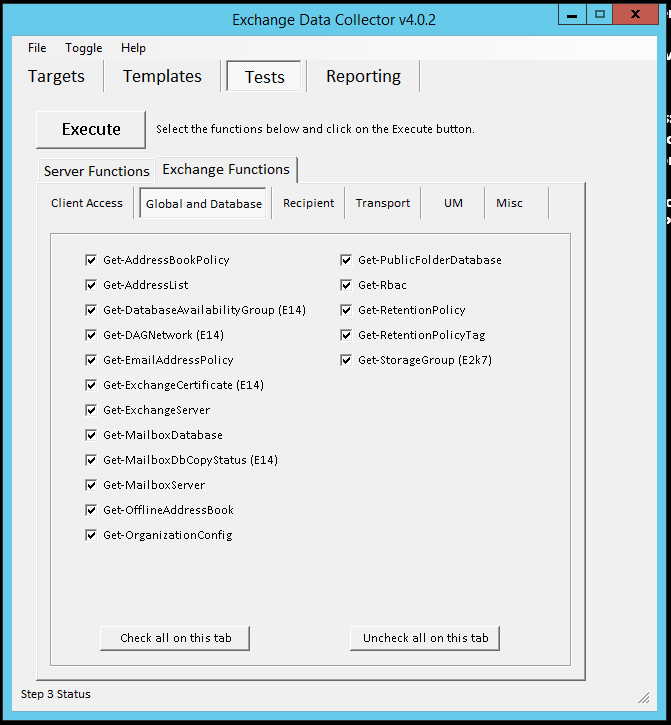
1. Select the Exchange Servers tab and insure that the desired functions for Exchange Servers are selected.



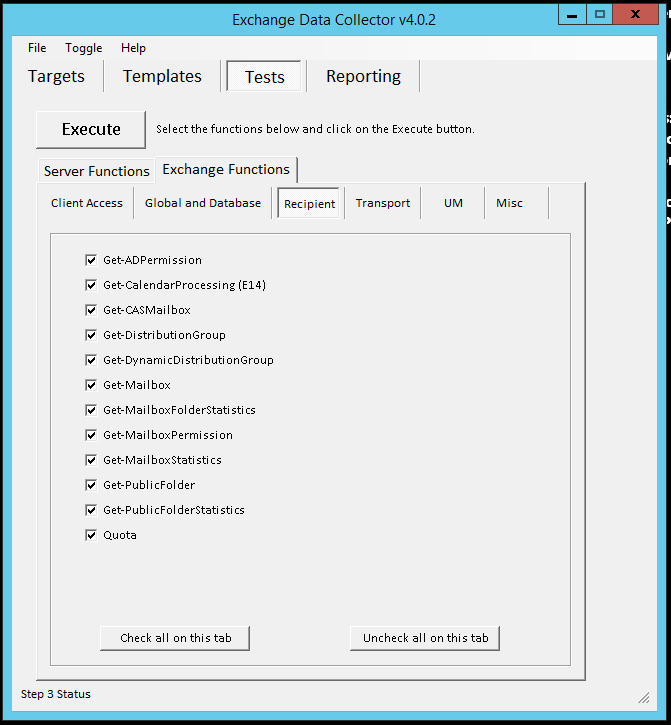
1. Select the Exchange Functions tab and insure that the desired Client Access functions are selected.



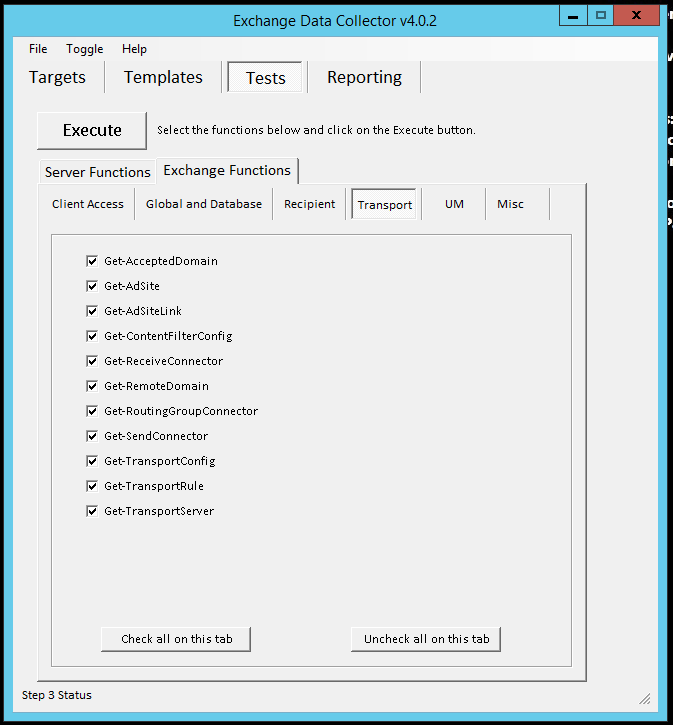
1. Select the Global tab, and insure that the desired functions are selected



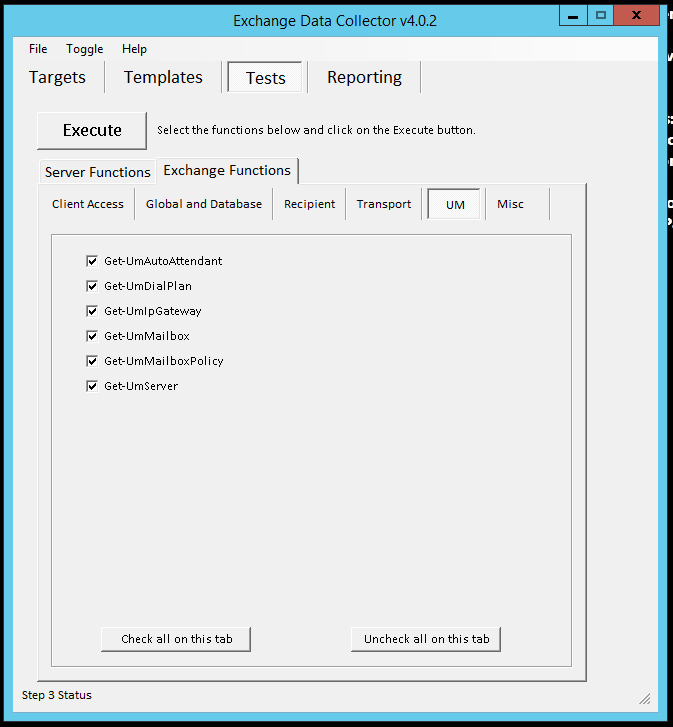
1. Select the Recipient tab and insure that the desired functions are selected.



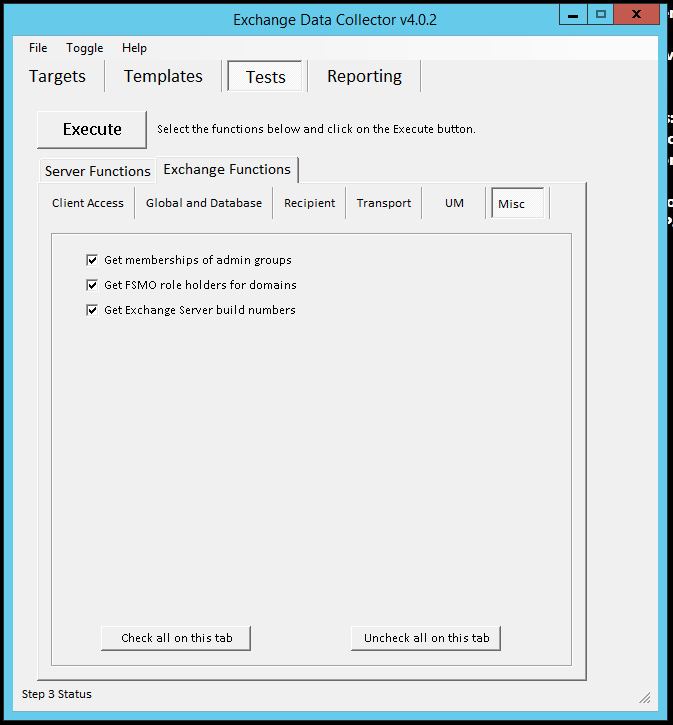
1. Select the Transport tab and insure that the desired functions are selected.



1. Select the UM tab and insure that the desired functions are selected.



1. Select the Misc tab and insure that the desired functions are selected.



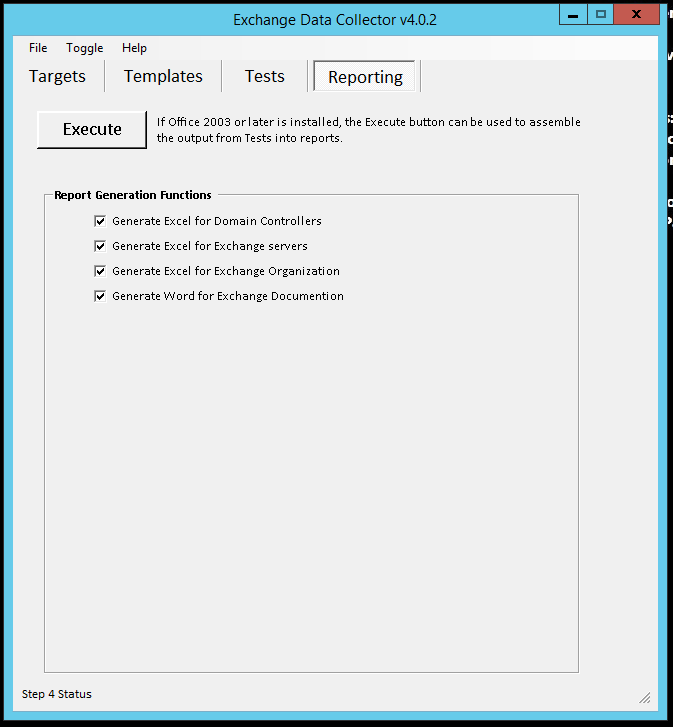
1. Once all desired servers and functions are selected, click the “Execute” button. The form will minimize, and progress can be tracked in the initial Powershell window. Once all jobs complete, the form should restore to its original location.

**Assembling ExDC Data into Reports**

Note: When the report generation completes, each of the spreadsheets will contain worksheets for all possible functions – regardless of whether data is present for the function. Worksheets without corresponding data will contain headers only. The Environmental Doc will also contain sections regardless of whether the data is present for the function.

### Scenario 1: Excel 2010 or later is installed on the Data Collection Server

1. After data collection completes, select the “Reporting” tab at the top of the window.



1. By default, all reports are selected. Any of these can be unchecked, if desired.
2. Once all desired reports are selected, click the “Execute” button. The form will minimize, and progress can be tracked in the initial Powershell window. Once all selected reports complete, the form should restore to its original location.

### Scenario 2: Excel 2010 or later is not installed on the Data Collection Server

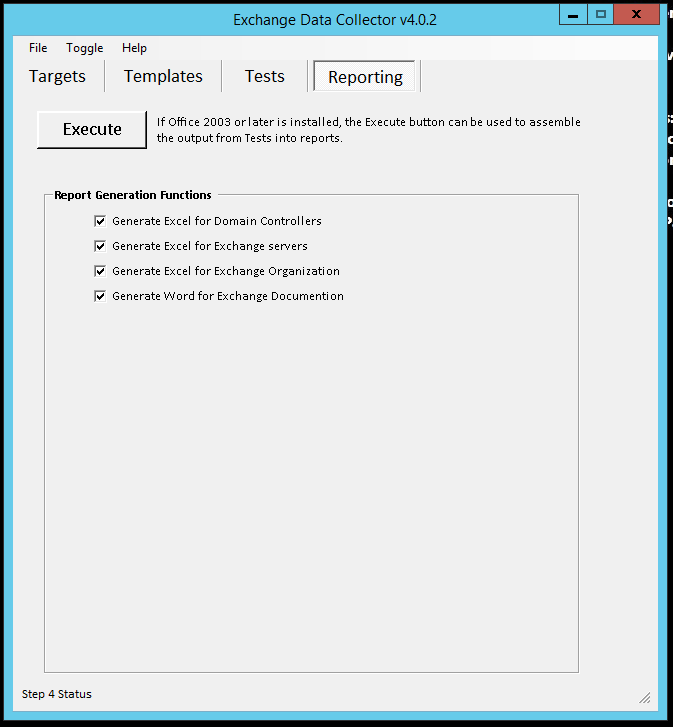
1. The entire working path folder should be moved to an alternate server or workstation with Excel and Powershell v2 installed. It is not necessary that this server or workstation have the Exchange Management Tools installed.
2. Open Windows Powershell v2
3. Set the execution policy to be unrestricted by typing:

Set-ExecutionPolicy Unrestricted

1. In the Powershell window, navigate to the working path of the ExDC files
2. Run ExDC with the following command:

.\ExDC.v4.ps1 -NoEMS

1. Once the form initializes, the three main tabs – Targets, Tests, and Reporting – will be visible at the top. Select the “Reporting” tab at the top of the window.



1. By default, all reports are selected. Any of these can be unchecked, if desired.
2. Once all desired reports are selected, click the “Execute” button. The form will minimize, and progress can be tracked in the initial Powershell window. Once all selected reports complete, the form should restore to its original location.

**Troubleshooting ExDC**

### Powershell Transcript

The start-transcript functionality is enabled when the Tests start running. This file is created in the working path folder and contains the output shown in the Powershell window. This file is useful for reviewing the progress at a high level but does not contain much troubleshooting information.

Note: If ExDC fails during the Tests phase of a run and causes the form to crash with an unhandled exception, Stop-Transcript must be manually run from the Powershell window before restarting a run.

### FailedJobs text file

The FailedJobs text file is created in the working path folder if any jobs fail during an ExDC run. This file is designed to be a quick reference for failed jobs, so that issues can be addressed and the jobs restarted. Like the transcript, this file is intended to be a reference and does not contain much troubleshooting information.

### RunningJobs text file

While an ExDC run is underway, the RunningJobs text file will be created. This file is updated at the polling interval and contains the following information:

* Job Name
* Job State
* Job Process PID
* Job Process time running

This file is designed to keep track of how long specific jobs have been running at any point.

### Application Log

The core of the troubleshooting for ExDC is contained in the application log. During the run, ExDC will add events with a source of “ExDC” to the application log on the data collection server. Events are logged:

* Event ID: 1
  + Information when ExDC starts
* Event ID: 2
  + Information when Exchange snap-ins are detected
* Event ID: 3
  + Information detected versions of Exchange
* Event ID: 4
  + Information the ExDC run parameters
* Event ID: 10
  + Information when “Discover” or “Load from File” is started on any Targets tab
* Event ID: 11
  + Information when “Discover” or “Load from File” completes on any Targets tab
* Event ID: 12
  + Information about schema version detected in Core\_Build\_Nodes.ps1
* Event ID: 20
  + Information when “Load from File” is started on any Targets tab
* Event ID: 21
  + Information when “Load from File” completes on any Targets tab
* Event ID: 30
  + Information when “Execute” is started on the Tests tab
* Event ID: 31
  + Information when “Execute” completes on the Tests tab
* Event ID: 35
  + Information when job completes. This event contains some data about the memory usage of the job as well as the process run time in seconds.
* Event ID: 40
  + Information when “Execute” is started on the Reporting tab
* Event ID: 41
  + Information when “Execute” completes on the Reporting tab
* Event ID: 42
  + Information when each Core\_Assemble\_\* script starts
* Event ID: 43
  + Information when each Core\_Assemble\_\* script completes
* Event ID: 49
  + Warning when “Execute” is started on the Reporting tab but Excel is not installed
* Event ID: 100
  + Error when Trap is hit. If the trap is hit in the job, this event should contain the details of the error encountered.
* Event ID: 400
  + Error when target fails ping test
* Event ID: 500
  + Error when job fails
* Event ID: 600
  + Error when job process is killed because job has exceeded the timeout value