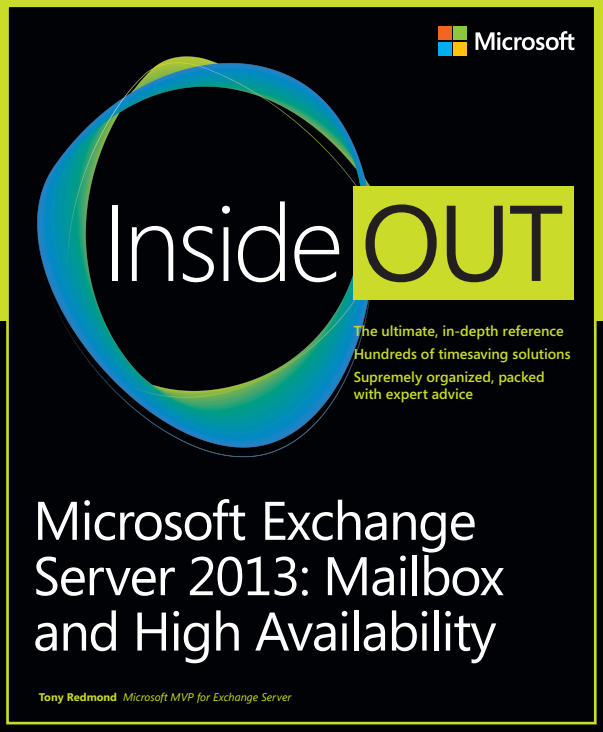
EXCERPT



Chapters 5-6

Managing Mailboxes, Groups, & Other Objects

Tony Redmond

Microsoft

Managing Mailboxes, Groups, & Other Objects: EXCERPT from Microsoft® Exchange Server 2013 Inside Out

Tony Redmond

PUBLISHED BY
Microsoft Press
A Division of Microsoft Corporation
One Microsoft Way
Redmond, Washington 98052-6399

Library of Congress Control Number: 2013948708
ISBN: 978-0-7356-8071-5

Printed and bound in the United States of America.

First Printing

Microsoft Press books are available through booksellers and distributors worldwide. If you need support related to this book, email Microsoft Press Book Support at mspinput@microsoft.com. Please tell us what you think of this book at http://www.microsoft.com/learning/booksurvey.

Acquisitions Editor: Anne Hamilton
Developmental Editor: Karen Szall
Project Editor: Karen Szall
Editorial Production: nSight, Inc.
Technical Reviewer: Paul Robichaux; Technical Review services provided by Content Master, a member of CM Group, Ltd.
Copyeditor: Kerin Forsyth
Indexer: Lucie Haskins
Cover: Twist Creative • Seattle

Contents at a Glance

Introduction

When Paul Robichaux and I first sat down to discuss how we might cooperate in writing Microsoft Exchange Server 2013 Inside Out, we were conscious that the sheer complexity and breadth of the product meant that many months of fact-gathering, writing, checking, and editing would be necessary to produce a book that described Exchange 2013 in sufficient depth and detail to warrant the "Inside Out" title. In fact, we knew that two books were necessary to avoid ending up with one 1,600-page mega-volume, so we divided the task to align with the Mailbox and Client Access Server roles, which is the plan that we've used to create the books.

At the same time, we knew that there were particular parts of Exchange 2013 that justified their own book. I've often felt that Unified Messaging was a very misunderstood part of the product. Paul has worked in this area for many years and has taught the subject to students aspiring to become Microsoft Certified Masters for Exchange. He's the best possible person to write about Unified Messaging. I have had a particular interest in compliance, and it's an area in which Microsoft has invested massively over Exchange 2010 and Exchange 2013, so it was easy (relatively) to create a mini-book on this topic. Then we were faced with High Availability, an area that is so important to so many companies who depend on their email being available all the time. The advent of the Database Availability Group (DAG) was a tremendously important advance for the product in Exchange 2010, and it's even better in Exchange 2013. Making High Availability the focus of the third mini-book made a lot of sense.

Each mini-book is a chapter from the larger "Inside Out" title, but each stands on its own merits and can be read in isolation. However, if you really want to get acquainted with Exchange 2013, you might just want to check out the two-volume set. We think you'll like it.

Errata & book support

We've made every effort to ensure the accuracy of this book and its companion content. Any errors that have been reported since this book was published are listed on our Microsoft Press site:

http://aka.ms/ExIOv1/errata

If you find an error that is not already listed, you can report it to us through the same page.

If you need additional support, email Microsoft Press Book Support at *mspinput@microsoft.com*.

Please note that product support for Microsoft software is not offered through the addresses above.

We want to hear from you

At Microsoft Press, your satisfaction is our top priority, and your feedback our most valuable asset. Please tell us what you think of this book at:

http://www.microsoft.com/learning/booksurvey

The survey is short, and we read every one of your comments and ideas. Thanks in advance for your input!

Stay in touch

Let's keep the conversation going! We're on Twitter: *http://twitter.com/MicrosoftPress*.

Paul is available on Twitter *@PaulRobichaux*. His blog is available at *http://paulrobichaux.wordpress.com/*. Tony is available on Twitter *@12Knocksinna*, while his blog is at *http://thoughtsofanidlemind.wordpress.com/*.

CHAPTER 5

Mailbox management

There's not much point in running an email system unless people can use it to communicate. Learning how to use the tools available to manage these objects is a fundamental first step in this process. Microsoft has justifiably been criticized in the past for making Exchange Server complex to manage, largely through the use of badly documented and archaic registry settings to enable or manage features. In addition, customers have asked for tools that are flexible enough to accommodate the needs of a range of administrative personnel, from local administrators who take care of tasks for just one server to help desk personnel to organization administrators. For example, in Exchange Server 2007, if you want to give someone the ability to maintain user properties such as the office in which he works, you must give that person access to the full-blown management console. In addition to the tools, granting and maintaining permissions to enable people to work with Exchange is complicated, resulting in the potential for error in granting permissions that could expose data to unintended manipulation by untrained users. Apart from the limited set actions that can be recorded through administrative auditing, there is no good way to capture exactly what happens to an Exchange server by using the administrative tools.

Using the administrative model as implemented in previous versions of Exchange caused administrators to spend far too much time doing mundane things to keep an Exchange organization healthy when they could be more productive elsewhere. Microsoft improved matters in Exchange 2010 with the introduction of the browser-based Exchange Control Panel (ECP), an advance that enabled administrators to manage Exchange on any device that could run one of the supported browsers, including tablet and smart phone devices.

As good as it was to see the introduction of ECP, it meant that Exchange 2010 boasted three management interfaces—the Microsoft Management Console (MMC)–based management console, ECP, and the management shell. Having three tools was confusing, especially when some options (such as group naming policies) were only available in ECP, and others (such as database management) only appeared in Exchange Management Console (EMC). The influence of Exchange Online also mandated a need for change because more web-based rather than Windows-based management is a higher priority as we head into an era of cloud-based services.

The upshot is that Exchange 2013 has rationalized down to two management interfaces. The Exchange Administration Center (EAC) is very much like ECP in that it is browser-based and presents different options through portions of the user interface (UI) that are revealed or suppressed depending on a user's membership in role-based access control (RBAC) groups. EAC is very different from the previous Windows-based administration consoles, and it takes a little while for an administrator to become truly comfortable working with the tool. Exchange Management Shell (EMS) underpins everything and remains the most powerful management tool because of the wider range of parameters you can adjust and how you can build scripts to manage Exchange the way you want rather than the way an engineer has designed.

Seeking perfection halts progress

It's easy to criticize the appearance and functionality of EAC by saying that EMC was so much better and more powerful. This statement is doubtful in any case, but the real point is that MMC-based consoles and the operating model they embrace, which is to log on to servers to perform management tasks, is rapidly becoming an outdated mode for Windows server administration, especially in large deployments. Microsoft has acknowledged that this is the case by incorporating so much potential for automation through Microsoft Windows PowerShell in Windows Server 2012.

Exchange adopted Windows PowerShell as the basis for management much earlier than any other Microsoft server application and then went on to introduce browser-based management. EAC combines the two procedures as shown in Figure 5-1. This console is different in terms of layout and capability from previous consoles, but it is capable of running on many types of devices, from Apple iPads to Surface RT tablets to Android smart phones. The parts of EAC that aren't quite as functional as previous consoles will improve through future software releases, just as any other piece of software improves over time. In the interim, EMS is always available to handle the most difficult and complex tasks.

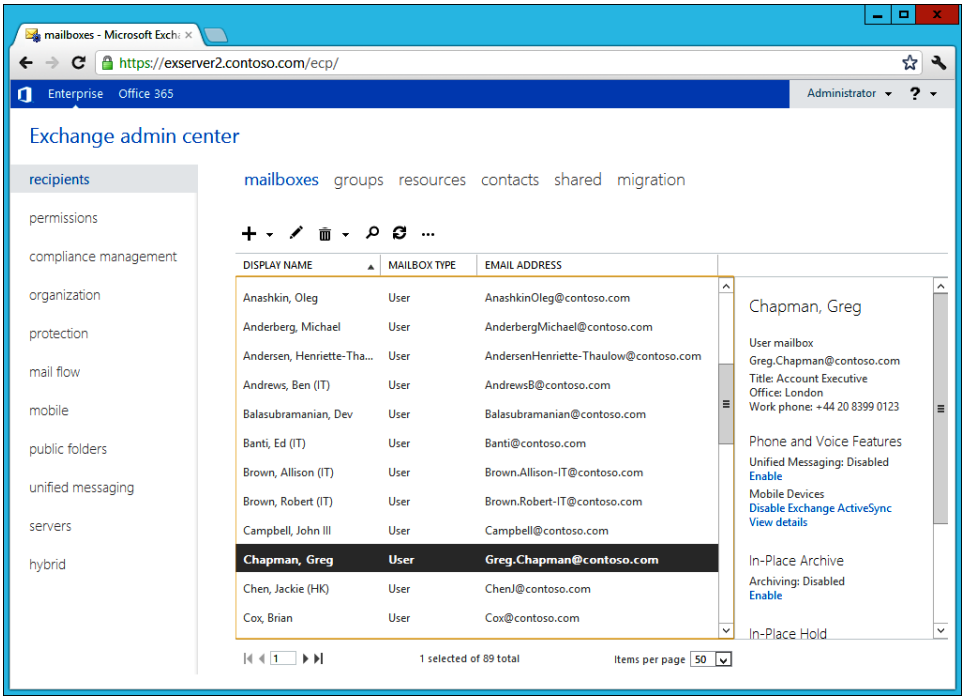


Figure 5-1 The Exchange Administration Center

Figure 5-1 shows how EAC organizes its view of management of the Exchange organization into the following major parts:

- **Recipients** Management of mailboxes, groups, contacts, room mailboxes, shared mailboxes, and mailbox migration (move).
- **Permissions** Management of RBAC-controlled administrative and user roles and Outlook Web App policies.
- **Compliance Management** Management of in-place hold (for mailboxes), discovery management, retention tags and policies, mailbox and administrator auditing, data loss prevention, and journaling.
- **Organization** Management of federated trusts for sharing with other organizations, third-party apps for use with Outlook Web App, and address lists.
- **Protection** Management of the Exchange 2013 anti-malware filter.

- **Mail Flow** Management of transport rules, delivery reports, accepted domains, email address policies, receive connectors, and send connectors.
- **Mobile** Management of ActiveSync quarantined devices, device access rules, and ActiveSync mailbox access policies.
- **Public folders** Management of modern public folders and public folder mailboxes. Management of traditional public folders is done through EMS or the Public Folder Management Console. You can't install or create traditional public folders if you haven't used them prior to the deployment of Exchange 2013.
- **Unified Messaging** Management of dial plans and IP gateways.
- **Servers** Management of Mailbox and Client Access Server (CAS), Database Availability Groups (DAGs), databases, virtual directories, and Secure Socket Layer (SSL) certificates.
- **Hybrid** Management of the connection between Exchange on-premises and Exchange Online (Office 365).

Like EMC, EAC is capable of operating in a multi-forest environment, assuming that the necessary trusts are in place to allow authenticated cross-forest access. For more information on this topic, see *http://blogs.technet.com/b/exchange/archive/2012/08/30/using-eac-to-manage-multi-forest-exchange-deployments.aspx*.

INSIDE OUT Some missing pieces of administrator functionality

Some functionality has been lost in the transition from EMC to EAC. A possibility exists that some will reappear in a future release or update that Microsoft issues for Exchange 2013. You might consider that losing any functionality means that EAC represents a step backward when compared to previous consoles and, on a feature-by-feature basis, this feeling is probably true. However, Microsoft will gradually whittle down the list of missing features and increase EAC functionality as it releases updates for Exchange 2013. For example, Exchange 2013 RTM CU1 reintroduced the ability for groups to manage groups. However, EAC offers better coverage in parts than other consoles. For instance, compliance management is generally better served in EAC than it is in the Exchange 2010 EMC, EAC includes the ability to import mailbox data to and export mailbox data from PSTs, and EAC includes alerts (see Figure 5-2) to notify administrators about important events. None of these options are available in EMC.

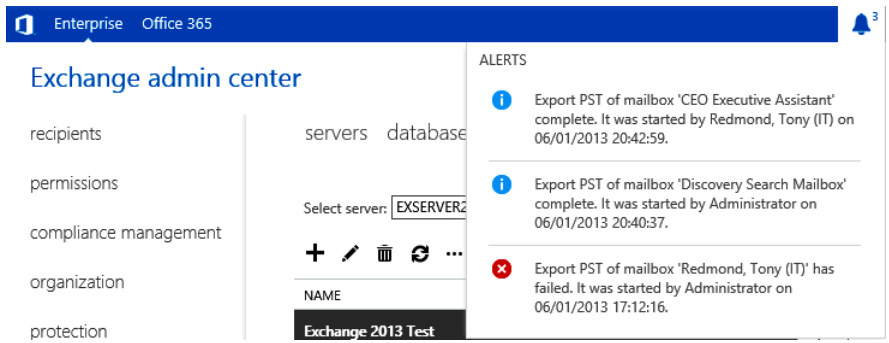


Figure 5-2 EAC Alerts

Many also prefer the way EAC presents information required to accomplish tasks because they think the EAC approach is easier to understand than the multistage wizard structure often employed by EMC. Building transport rules is one example of when EAC is arguably easier to understand than EMC.

Like EMC, there are always parts of the administrative task list that can't be handled through an option presented through the graphical user interface (GUI). These tasks are usually in the "rare and uncommon" category and require a reasonable degree of product knowledge before they can be addressed. For example, there's no option to view mailboxes the Store has quarantined because of a suspected corrupt item. Likewise, no option exists to run the Store fix-up-in-place cmdlets to resolve minor logical corruptions in a database's tables. You might never find options for these tasks presented through a GUI, if only because it's a way of underlining that you really need to know what you're doing when you take risks.

Starting EAC

Like EMS, EAC is based on remote Windows PowerShell and RBAC. When EAC starts, it contacts the server specified in the URL you typed into the browser to initialize a remote PowerShell session, for example, *https://exserver1.contoso.com/ecp*. You can see that EAC connects to the ECP virtual directory to give the same URL as used with the Exchange 2010 Control Panel. This is to provide a certain degree of backward compatibility. If you type **https://exserver1.contoso.com/eac**, you get an HTTP 404 error because Internet Information Services (IIS) has no EAC virtual directory to which it can connect.

The URL for EAC can have the relevant section appended to bring you to a particular location. For example, *https://exserver1.contoso.com/ecp/?exsvurl=1&p=Mailboxes* starts EAC and positions the console in the Mailboxes section, whereas *https://exserver1.contoso.com/ecp/?exsvurl=1&p=Mobile* starts EAC in the Mobile (ActiveSync) section.

INSIDE OUT Making sure that you see the right EAC

When you make a connection to EAC, Exchange queries Active Directory to discover the version running on the server that hosts your mailbox to ensure that the correct UI is displayed. This means that if your mailbox is still on Exchange 2010, you will see ECP rather than EAC. To force Exchange to display EAC, you need to add *?ExchClientVer=15* to the URL. For example: *https://server/ecp/?ExchClientVer=15*.

After EAC has established a remote Windows PowerShell session, it begins to retrieve the data necessary to fill in whatever part of the UI is selected. During initialization, EAC executes the cmdlets RBAC permits (for the account used to run EAC) to discover information about the organization, servers, and so on to build its cache with essential data about the Exchange organization. Later, EAC executes other cmdlets to retrieve information about specific objects as the user navigates from node to node. For example, if the user moves to Recipients and clicks Mailboxes, EAC runs Get-Mailbox to fetch the information to display. Because of the way it fetches and caches data, EAC performs better than EMC when dealing with large amounts of information, such as fetching details of 2,000 mailboxes. In this respect, EAC works in a similar manner to Outlook Web App when it navigates through mailbox folders that contain thousands of items.

RBAC ensures that an administrator sees only the options with which she can work. For instance, if your account doesn't hold the Discovery Management role, you might be able to create a new in-place search, but you cannot edit the query Exchange uses to locate items in user mailboxes. In other words, a regular administrator can set up the framework for a search, but the search can be activated only to collect items from user mailboxes by a user who is a member of the Discovery Management RBAC role group.

Likewise, even if your account holds the Organization Management role and is therefore able to see just about every option imaginable, you won't see the options to import or export mailbox data unless an explicit role assignment has been made to assign the Mailbox Import Export role to your account. Building a customized UI based on a user's role is a good thing because it stops frustration caused when someone attempts to take an action that he doesn't have the necessary permission to perform, even if it creates some new questions when users ask why their version of EAC is different from someone else's (or from what they read about in books or online materials). However, when loaded during the EAC initialization process, the RBAC data is inflexible in that if a change is made to a user's role, EAC will not reflect the change until the next time it loads and rebuilds its cache by reading RBAC information from Active Directory.

Unlike Exchange 2010, in which you have to install a language pack to use the management tools in different languages, Exchange 2013 installs the necessary language-dependent pieces to enable administrators to manage Exchange in their preferred language. When EAC starts, it uses the locale determined by the language setting in the user's mailbox. For example, if your language setting is en-us, you see the U.S. English version of EAC, whereas if it is fr-fr, you see the French (Figure 5-3). The language setting is populated the first time a user runs EAC or Outlook Web App or by running the Set-Mailbox cmdlet to write a value into the languages property. How to control the language setting for users is discussed in the "Languages" section later in this chapter.

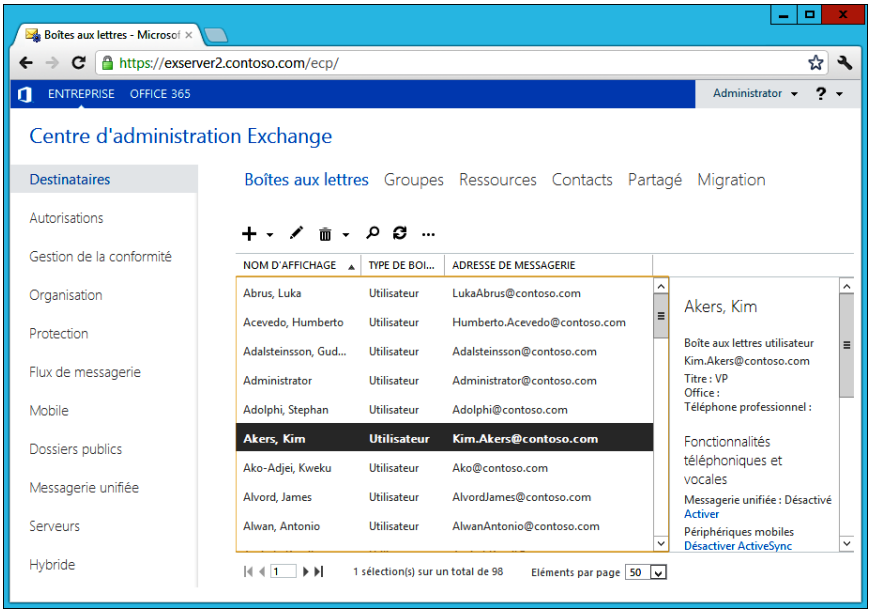


Figure 5-3 Running EAC in French

You see only the tasks allowed for your role

Like EMS, EAC uses RBAC to ensure that users see only the tasks their role allows them to perform. In other words, EAC can modify the options it displays to reflect the roles a user holds. Administrators who hold the Organization Management role see all options and can work with user data; administrators who hold a more restricted role see a limited set of options. EAC also respects any RBAC scopes that are in place to restrict users to working with a specific set of servers or databases.

How EAC accesses Exchange data

EAC has an absolute dependency on Active Directory. Throughout a session, EAC reads and writes data about objects it fetches from the Microsoft Exchange organization container in the Configuration Naming context. Other data (such as the current quota a mailbox uses) comes from mailbox databases, but the vast majority of information EAC displays is sourced from Active Directory. However, you should never manage Exchange objects through the Active Directory tools because these tools possess an incomplete knowledge of objects such as Dynamic Distribution Groups.

The information contained in Active Directory is static and is not intended to reflect information that changes in real time because this would generate constant replication requests to keep pace with status updates for Exchange objects. Even if Active Directory could keep up with the replication, it's likely that the activity would swamp networks and prevent other useful work from being done. The dependency on Active Directory is why EMS sometimes exposes transient data that you never see in EAC. For example, you can use the Get-MoveRequestStatistics cmdlet to view the percentage of a mailbox move that is complete and the current rate of data transfer between source and target server, but you never see this level of detail in EAC. Instead, EAC displays the status of the move requests in a migration batch from start to in progress to complete, but only if you refresh the display.

INSIDE OUT **Getting the latest information**

Because EAC essentially gives only a static snapshot of the set of objects at which you are looking, it is wise to use the refresh option before you start to do anything with EAC to make sure that you are dealing with the latest information rather than stale data. For instance, refreshing the set of mailboxes picks up new mailboxes that have been added and updates mailboxes that have had status changes, such as those that are being moved to a different server or those that have just been given an archive.

Note

When you view mailbox properties, you might see that a mailbox has been last logged on to by an account that doesn't own the mailbox. This occurs when another user has logged on to the mailbox by using delegated permissions that he has been granted.

Unlike EMC in Exchange 2010, which enables you to select a specific domain controller from which the console fetches Active Directory data, EAC automatically selects a domain controller from the set available in its local site and doesn't allow you to change this server during a session. Another difference between EMC and EAC is the way the two consoles fetch large numbers of objects. For performance reasons, EMC limits itself to fetching 1,000 objects unless explicitly forced to fetch more. The default value worked well for small installations but was not so good when large numbers of objects existed, such as the number of mailboxes supported by large organizations. EMC got over this difficulty by allowing administrators to modify the maximum number of objects to be fetched. Clearly, it took longer to fetch 10,000 objects than 1,000, but it was a reasonable solution. EMC also supports extensive filtering capabilities to enable an administrator to view a subset of objects, such as all the mailboxes located in a certain database or those that belong to a specific department.

EAC takes a completely different approach to fetching objects and uses a similar mechanism to the way Outlook Web App navigates through mail folders. A folder in an Exchange mailbox can hold tens of thousands of items, so it's unreasonable to expect any client to fetch all items when it opens a folder. Outlook Web App navigates on a page-by-page basis so that it fetches sufficient data to display enough objects to fill the current screen, which enables the user to begin working with data, and then fetches enough data to populate the next few screens to support the user navigating further within the folder. Outlook Web App can also move quickly between the top and bottom of a folder. In the case of EAC, the UI is designed to move through data by using 50 objects per page, but you can adjust it to display 100, 200, or even 500 objects per page to accommodate larger screen sizes. Behind the scenes, EAC caches more data so that you can move from page to page quickly. This approach limits the amount of data that has to pass between server and client while also enabling the UI to perform well, even when confronted with large quantities of data.

Changing EAC columns

Like EMC, the recipients section of EAC can be customized to add or remove columns to make the data shown more useful to an administrator. Click the ellipses and choose Add/Remove Columns to change the columns you see when you access different types of objects. Figure 5-4 shows the process in action. EAC does not offer the same facility for the other sections within the console.

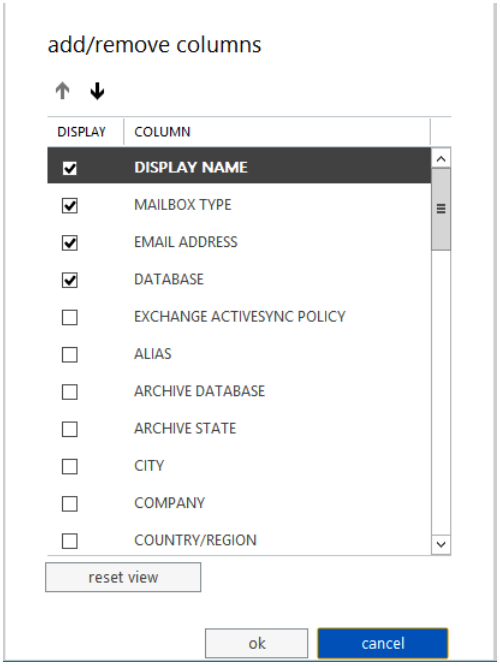


Figure 5-4 Selecting columns for EAC to display

Naming conventions

The topic of naming conventions should be covered during your planning for deployment. Server naming conventions was discussed in Chapter 2, "Installing Exchange 2013." It's clearly important for servers to be assigned names that make sense and convey some information about a server's purpose; this makes it much easier for administrators to manage the organization. Other important objects that deserve some attention in naming include the following:

- **User mailboxes** My strong preference for many years has been to use the Last Name, First Name convention because the convention mimics the way old-fashioned telephone directories work, so it's easier to navigate the large groups of users in the Global Address List (GAL) who share common surnames (such as Smith or Ng). The convention also works well for multinational companies that have to accommodate non-European surnames.

- **Room and equipment mailboxes** Most companies have already named rooms in buildings so it makes sense to follow the established convention. When an organization includes rooms in multiple buildings, you might want to prefix the room name

with a building identifier. For example, the Frankfurt conference room in Building 43-1 might be called B43-1-Frankfurt. Building names tend to be well understood by users, so you can afford to be a little cryptic in the names for these mailboxes.

- **Groups** Ideally, general-purpose distribution groups should convey the use of the group (for example, Exchange 2013 Interest List), and those intended for business-linked communication should indicate the business group and purpose (for example, Finance Department Planning Group). Common sense and consultation with the group owners to understand the purpose of the group usually leads to a sensible and easily understood name. Some companies, including Microsoft, add an indication to show users when a group is based on a query so that they won't bother looking up the directory to check group membership. Microsoft appends (QBDG) to the ends of these group names, so you end up with a group name such as All Users (QBDG). Exchange enables you to implement a group naming policy, which is described in Chapter 6, "More about the Exchange Administration Center."

- **Mail-enabled contacts** These objects should use the same naming convention as user mailboxes.

- **Public folders** Use the same approach to naming as for distribution groups. Above all, avoid any temptation to be cryptic because it can be hard enough to navigate the public folder hierarchy without creating another obstacle to user comprehension.

- **DAG** These objects are visible to administrators only, but it's still important to use a convention that informs administrators about the DAG's purpose.

- **Databases** Exchange 2010 and Exchange 2013 require databases to have names that are unique within the entire organization. The simplest convention is to assign names that indicate what mailboxes exist in the database. This could be the department name if you group mailboxes by department. Some companies indicate the mailbox size in the name so that the administrators know where to put mailboxes of a particular type and size when they are created. For example, UK Sales-1GB indicates users who belong to the U.K. sales department who have 1 GB mailboxes. Descriptive database names certainly work, but it becomes more difficult to think of good names to use after you have more than 20 or 30 databases to manage. See Chapter 9, "The Database Availability Group," for a more comprehensive discussion on how to name databases in large-scale deployments.

- **Connectors** Messaging connectors should have names that clearly indicate their purpose and the type of traffic they support, for example, SMTP to Internet or SMTP to Lotus Notes.

INSIDE OUT **Avoid retroactive naming policies**

Don't create a heap of objects and then attempt to apply a retroactive naming policy; it is dreadfully boring to have to go through objects to rename them. Take the time early on to decide on a naming convention and then communicate the convention with some examples to any administrator who has the permission to create objects in the organization.

Managing Recipients

Recipients cover the broad spectrum of any object that can receive mail, or mail-enabled objects. EAC groups four major types of these objects under Recipients. These are:

- Mailboxes
- Groups
- Resources (room and resource mailboxes)
- Contacts (including mail users)

The following sections discuss each of these recipient types and their management.

Recipient filtering

An email server such as Exchange deals with huge differences in terms of object numbers. The largest Exchange organizations running outside Office 365 have over half a million mail-enabled objects. However, the smallest organizations might deal with just a few dozen mailboxes. EAC is designed to handle anything from the smallest demonstration environment to the largest production deployment.

Clearly, only a limited amount of information can be presented on any computer screen. As it moves to the different types of recipients, EAC displays the initial set of objects in alphabetical order. To see different sets of objects, you must either scroll down through pages of data (tiresome if the object you want is a number of pages down) or use the EAC search capabilities.

EAC does not support the same kind of recipient-filtering capabilities as does EMC, which allows the console to focus on specific sets of objects such as all the mailboxes in a specific database. Instead, all the sections of EAC that might deal with large numbers of objects, including recipients, have a search box in which you can type some characters to request

EAC to display matching objects. Figure 5-5 shows how the search box works when dealing with mailboxes. In this instance, EAC has detected that you might be searching against the various name attributes stored for mailboxes. Not all mailbox properties can be used for searching. For example, you cannot search against the company attribute, but you can against the department.

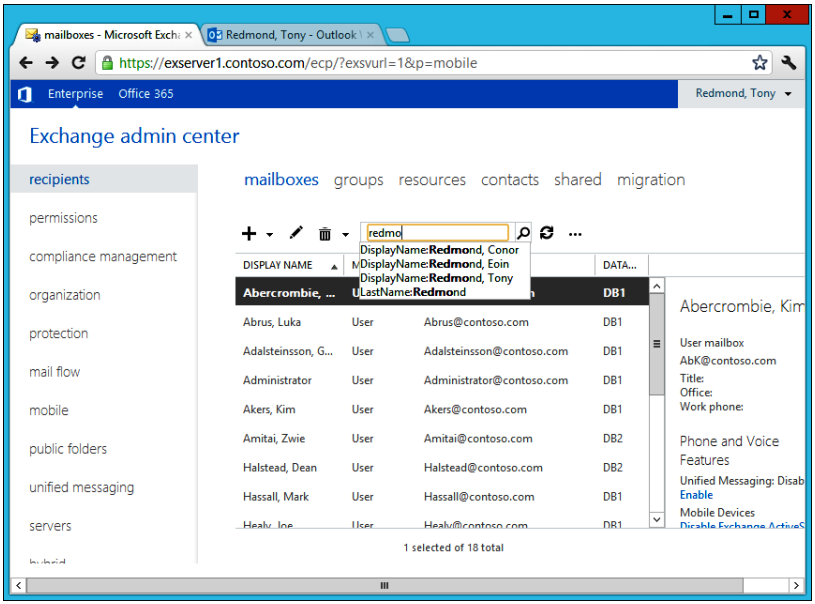


Figure 5-5 Searching for specific mailboxes in EAC

EAC also includes an Advanced Search option available by clicking the ellipsis. This option is available for all recipient types and can be used to look for any recipient of the selected type (mailboxes, groups, contacts, public folders), using values such as those stored in the 15 custom attributes available for Exchange mail-enabled recipients. Advanced Search is the closest to the kind of filtering that EMC offers and is just as powerful.

Exporting EAC information to CSV files

EAC includes an option to export the current list view to a comma-separated values (CSV) file. CSV files are often used to work with Exchange data because they can be loaded into applications such as Microsoft Excel and Microsoft Access and then manipulated before being reused for some purpose. For instance, the standard method used by Exchange 2013 to move mailboxes is to create and process migration batches (see Chapter 10, "Moving Mailboxes," for more information). You can input a set of mailbox names to be moved, or you can provide EAC with a CSV file containing the set. Say you wanted to move all the mailboxes that are currently in database DB2 to a new database. To do this, you'd specify

database = DB2 in an advanced search. After you execute the search, EAC displays all the mailboxes in the database. You can then click the ellipsis and choose to export the current view to a CSV file. You then select the fields to be exported (only the email address field is necessary for a migration batch) and click Export. EAC generates a CSV file you can open with Notepad (Figure 5-6).

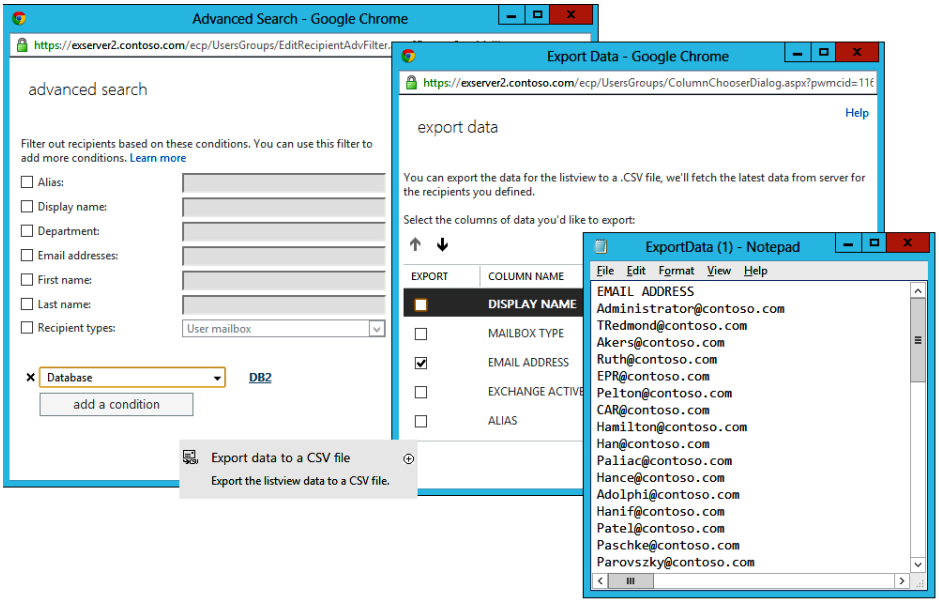


Figure 5-6 Building a CSV file from mailbox information

This is a very simple example of how mailbox information can be exported and reused within Exchange 2013. You can export information to a CSV file by using similar steps for any of the object types that are managed through the recipients section.

Some mysterious mailboxes

Although they exist, EAC does not reveal the existence of system mailboxes. In this respect, you have:

- Arbitration mailboxes, including the discovery search mailboxes (see the "Discovery mailboxes" section later in this chapter). Exchange uses arbitration mailboxes for many purposes, including message moderation. To see details of these mailboxes, you execute the Get-Mailbox –Arbitration command.

- Mailboxes created to test and measure system health. Exchange creates these for use by the Managed Availability system to test that different components of the

system are working correctly. For example, Exchange uses health mailboxes to send messages to each other to verify that the transport system is working as expected. You can see the set of health mailboxes within an organization by running the Get-Mailbox –Monitoring command.

- Mailboxes created for test purposes. For example, the New-TestCASConnectivityUser .ps1 script provided with Exchange creates a mailbox called extest_867e89ec8f7b4.

These mailboxes have a clearly intended purpose; otherwise, Exchange would not create them. For this reason, you should not delete or otherwise change their Active Directory accounts or mailbox details. The exception to this rule is when you want to remove a mailbox database that contains some arbitration or discovery mailboxes. In this case, you always need to move these mailboxes out of the database before you can remove it.

The need for mailboxes

Although it might be nice to be able to boast about the sheer scale of the organization you manage when you meet your peers at events such as the Microsoft Exchange Conference, managing a successful Exchange deployment is not determined by the number of mailboxes in the organization. Before you rush to create any mailboxes, you should lay out some guidelines for when mailboxes are created and when they are removed. Best practice for mailbox maintenance includes the following important points:

- Applications don't need mailboxes. Some administrators assume that it is a good thing to assign mailboxes for use by applications that need to create and send messages, usually by submitting a text message to an SMTP server. Applications do not need mailboxes for this purpose because they can create and submit messages to an SMTP server that supports submission from anonymous senders. The easiest way to support email submission for applications is to use the transport pickup directory. (For more information, see *Microsoft Exchange Server 2013 Inside Out: Connectivity, Clients, and UM*, by Paul Robichaux [Microsoft Press, 2013], Chapter 2, "The Exchange transport system.") If you do create mailboxes for application use, make sure that you secure the accounts associated with the mailbox so that they are restricted.

- Mailboxes have different types for a reason. Although it might seem fine to use normal mailboxes for resources (rooms and equipment), Exchange has a purpose behind the differentiation that it supports across mailbox types. Resource mailboxes are tied to disabled Windows accounts, and user mailboxes are not. Site mailboxes also use disabled Windows accounts. When you start to use normal mailboxes for resources, you create a potential security issue. Always assign the right mailbox type when you create a mailbox.

- Mailboxes shouldn't be kept forever. The information in a mailbox belonging to someone who leaves the company is probably of some interest, but interest wanes over time, and the information contained in most mailboxes belonging to an employee who has recently left is probably useless after three months. There will be exceptions, including mailboxes belonging to executives, which might be needed if an eDiscovery search is required for local information to respond to a legal action. Nevertheless, you should agree on guidelines to govern when mailboxes can be removed and make sure that old mailboxes and old Windows accounts don't linger past their best-by date (see Chapter 11, "Compliance").

> **Note**
>
> **Apart from anything else, old mailboxes and accounts could represent a security weakness that an attacker can exploit. Some companies move all mailboxes belonging to departed employees to a special database so that they are grouped and are obviously different from live mailboxes.**

Audit mailboxes regularly. You don't want to pay Microsoft any more for Client Access Licenses (CALs) than you should. CALs are often calculated on the basis of mailbox numbers, so it follows that keeping unnecessary mailboxes costs money. You should audit the mailboxes that exist in the organization at least every six months and remove any unused mailboxes. It's easy to report the last time a list of mailboxes in a database were logged on to detect potentially unused mailboxes. For example, the following command fetches details of all user mailboxes in a database and sorts them according to the last time a user logged on to the mailbox. Users who have never logged on to their mailbox are indicated by an error when Get-MailboxStatistics attempts to retrieve information from the mailbox. Other information that might indicate an unused mailbox, such as the total number of items in the mailbox, is also included. This report shows that approximately two months separate the most recent logon (my mailbox) from the oldest. It's reasonable to suspect that the mailboxes that have not been accessed in two months are no longer needed, or at least that they can be marked as being suitable for potential deletion if not required for regulatory purposes.

```
Get-Mailbox –Database DB2 –RecipientTypeDetails UserMailbox | Get-MailboxStatistics
| Sort-Object LastLogonTime | Format-Table DisplayName, LastLogonTime, ItemCount,
TotalItemSize
```

With these points in mind, you can create and manage some mailboxes.

Naming mailboxes

Email address policies enable you to define and apply different patterns for the SMTP addresses that are assigned to mail-enabled objects. The application of address policies makes sure that the SMTP addresses are consistent throughout the organization. Exchange 2013, unfortunately, does not provide a mechanism to control the generation of display names, which is the attribute that Exchange uses to sort objects in the GAL and EAC and for recipients and authors in message headers.

Table 5-1 lists the different attributes for the various names or name components Exchange uses that are stored in Active Directory. The default pattern for display names is %g %s; in other words, first name<space>last name or, in my case, Tony Redmond. This is an acceptable naming convention for small implementations in which everyone knows everyone else, and it is easy to find the correct recipient by browsing the GAL, but it becomes increasingly difficult to find people as the number of directory entries increases. The question, therefore, is what naming convention to use that is efficient and logical for users when they search for an object in the GAL. More variation occurs in surnames than in given names. Common given names, such as John or Mary, occur thousands of times in a large GAL, so if the GAL is sorted by given name, you might have a tiresome search before you locate the right recipient. It is easier to search using a surname, even with common surnames such as Smith, Chan, or Ng. Telephone directories are organized by surname, so it makes sense to carry the analogy forward and do the same thing for the GAL.

TABLE 5-1 Mailbox attributes and names

Attribute	Meaning
Alias	Unique name for the object
Name	Full name of the object composed of first name and last name
FirstName	First name of the user
LastName	Surname of the user
DisplayName	Name used to sort the GAL and for other display purposes (such as EAC and in message headers)
DistinguishedName	Name used to identify object in Active Directory
PrimarySMTPAddress	Primary SMTP email address (often first name.last name@domain)
UPN	User Principal Name, or the name of a user in email format that can be used to log on to a Windows server; recommended that the UPN has same value as the primary SMTP email address

INSIDE OUT Applying a different naming convention

Although the sequence used in naming conventions is hard-coded in EAC, it is possible to alter a convention. If you want to apply a different naming convention, the usual approach is to:

- **Allow EAC to create the mailboxes and contacts as normal and subsequently edit the Display Name.**
- **Create mailboxes and other mail-enabled recipients by using EMS scripts so that you have complete control over the format used for display names.**

There might be other circumstances in which you have mailboxes for which you don't want to use the last name, first name convention, such as those used for discovery results, but these can be dealt with on an exception basis. Figure 5-7 shows how Microsoft Outlook 2013 displays a GAL in which the mailboxes are organized using the last name, first name convention. As you can see, some of the entries have additional information to identify individuals who share common names or who have particular functions within the company. It is common to use department names, locations, or job titles to help users identify the correct recipient.

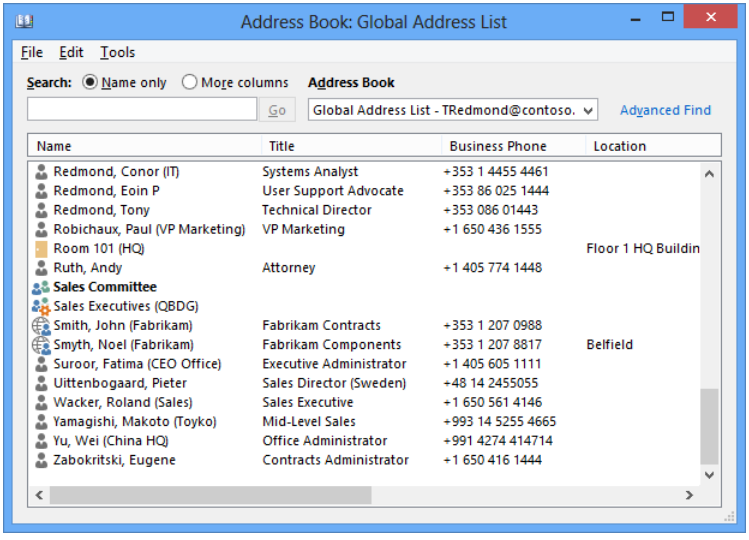


Figure 5-7 A well-ordered GAL

One problem with adding some detail to display names that deserves consideration is that it might expose some confidential information outside the company. For instance, assume that you have two people named John Smith in the organization, and you want to help users locate the correct person in the GAL, so you create display names that identify the two individuals by adding the department in which they work in parentheses. Thus, you might have:

- Smith, John (Corporate Acquisitions)
- Smith, John (IT Standards)

These names help users know which of the two John Smiths to whom they should address messages. However, the issue arises when the names of the departments are also communicated to recipients outside the organization. Anyone who receives a message from either John Smith will know for which department he works. This might not be important for generic departments such as Sales or Marketing or locations such as Dublin or New York, but it could be for departments such as Corporate Acquisitions or Talent Management, both terms that convey a lot about the role the user holds within the company. The lesson here is that you need to think about whether it matters if people outside your company know the information you add to display names to identify people in the GAL.

Some companies like to impose a special naming convention for distribution groups so that users know when they are sending a message to a group rather than to an individual recipient. The Exchange MailTips feature helps here; it can either warn users when they address a message to a large group or display a tailored tip to indicate the purpose of the group. One solution is to prefix groups with some characters. For example, you could create a group naming policy, DG, as a prefix so that your groups would have names such as DG: Sales Executives and DG: IT Department. The advantage of this approach is that all the groups are found in a single location in the GAL. Some take the idea further and use a prefix such as ## that places all groups at the start of the GAL. You then have names such as ## Sales Executives. This approach works but is not as user friendly as the other one.

Creating new mailboxes

Creating a new mailbox with EAC is easy. Open Recipients, select Mailboxes, and click the + (plus) sign to expose the dialog box to collect details about the new mailbox (Figure 5-8). Exchange 2013 supports the following mailbox types:

- **User mailboxes** The standard full-function mailboxes used by people to send and receive email and work with calendar, contacts, and other mail-enabled applications. User mailboxes can be associated with archive mailboxes to provide secondary longer-term storage. Archive mailboxes are discussed in Chapter 10.