

THE EXPERT'S VOICE® IN EXCHANGE

Pro Exchange 2013 SP1 PowerShell Administration

For Exchange On-Premises and Office 365

*GET GRANULAR CONTROL AND EASY
MANAGEABILITY WITH MICROSOFT'S
PREMIER HOSTED EXCHANGE PLATFORM*

Jaap Wesselius and Michel de Rooij

Apress®

For your convenience Apress has placed some of the front matter material after the index. Please use the Bookmarks and Contents at a Glance links to access them.



Apress®

Contents at a Glance

About the Authors	xix
About the Technical Reviewers	xxi
Acknowledgments	xxiii
Introduction	xxv
■ Chapter 1: Introduction to Exchange 2013 SP1	1
■ Chapter 2: Installing Exchange Server 2013	43
■ Chapter 3: Exchange 2013 Client Access Server	117
■ Chapter 4: Exchange 2013 Mailbox Server	151
■ Chapter 5: High Availability	229
■ Chapter 6: Message Hygiene	271
■ Chapter 7: Backup, Restore, and Disaster Recovery in Exchange Server 2013	305
■ Chapter 8: Unified Messaging	355
■ Chapter 9: Compliance	393
■ Chapter 10: Security	489
■ Chapter 11: Office 365 and Exchange Online	527
Index	601

Introduction

The Microsoft UC solutions are changing rapidly, especially from on-premises operations to cloud solutions. This makes it hard to write a book about Exchange 2013. Changing the subject to PowerShell Administration makes it a bit more version independent, but even during this writing, we found the applications were changing rapidly. The book is based on Exchange 2013 SP1, but you'll find some information regarding subsequent cumulative updates, as these are released on a quarterly basis.

This book is aimed at IT Pro's, the Exchange administrators with a couple of years experience who need guidance in deploying and managing Exchange Server 2013 on-premises, especially when it comes to managing PowerShell. Inside these pages are eleven chapters that cover the following topics:

- Chapter 1 - Introduction to Exchange 2013. This is an overview of Exchange Server 2013, including new and discontinued features, integration with Active Directory, and an architectural overview of the product.
- Chapter 2 - Installing Exchange Server 2013. The first part of this chapter covers the installation of Exchange Server 2013, both on Windows Server 2008 R2 and on Windows Server 2012 R2. The normal graphic setup is discussed; also here is the unattended setup with all the command-line switches that are available, including the post-installation configuration options. The second part includes information regarding coexistence with Exchange 2010 or Exchange 2007.
- Chapter 3 - Client Access Server. This covers the Client Access server, Client Access technologies, and all available clients for use with Exchange 2013. Namespaces, SSL certificates, load balancing, and Publishing Exchange 2013 Client Access server are the most important topics here.
- Chapter 4 - Mailbox Server. This covers the Mailbox server, including the available recipients like mailboxes, distribution groups, and public folders, as well as how to manage them. Except for mailboxes, another important part of the Mailbox server is message transport, which is also covered here.
- Chapter 5 - High Availability. High Availability is an important and complex aspect of every Exchange 2013 deployment. This chapter covers the basics of the database availability group and how to build and configure it. The chapter covers also Client Access High Availability and Transport High Availability.
- Chapter 6 - Message Hygiene. This topic is new in Exchange 2013 SP1, and so the chapter discusses the Edge Transport server and how to implement all available anti-spam features, plus how it integrates with the Exchange 2013 Mailbox server.
- Chapter 7 - Backup, Restore, and Disaster Recovery. This discusses backup technologies and how backup technologies interact with Mailbox database technologies. Other topics in this chapter are restore technologies and the Microsoft Native Data Protections, sometimes also referred to as "backup-less environment."

- Chapter 8 - Unified Messaging. This explores the UM feature set, shows how to configure Exchange UM for supported IP telephony solutions, and explains how to integrate Exchange UM with Lync. Other topics are UM mailbox policies, UM auto attendants, Unified Contact Store, call answering rules, and voice-mail preview.
- Chapter 9 - Compliance. This discusses the compliance-related features of Exchange 2013, such as in-place archiving to manage the primary mailbox, in combination with message records management and in-place discovery that, in conjunction with in-place hold, can be used to support legal investigations or other purposes. Data loss prevention and fingerprinting are discussed as features to prevent data leakage. Other topics are administrator and mailbox auditing.
- Chapter 10 - Security. This explores the role-based access model and all its components, such as management roles, scopes, role groups, and special-purpose features like unscoped top-level management groups. Other topics are the split-permissions model for organizations with separated management of Active Directory and Exchange, and S/MIME.
- Chapter 11 - Office 365 and Exchange Online. This shows how to connect to Office 365, and discusses Autodiscover, as well as how to federate organizations to share information such as calendaring. It also covers how to configure directory synchronization with Azure Active Directory and how to configure Active Directory Federation Services and Multi-Factor Authentication. Additionally, it explains how to move mailboxes between on-premises and Office 365, Exchange Online Archiving, and how to reconfigure mail flow when using Exchange hybrid.

CHAPTER 1



Introduction to Exchange 2013 SP1

In October 2012, Microsoft released the eighth major version of its messaging and collaboration server, Exchange 2013. This version of Exchange 2013 is referred to as *Exchange 2013 RTM*, the lattermost which stands for “Release To Manufacturing.” In early 2014, this release was followed by the release of Exchange 2013 Service Pack 1 (SP1). As usual, SP1 brings a lot of hotfixes, but also a lot of new features, both completely new features and features (including some that were available in Exchange 2010) that did not make it into Exchange 2013 RTM, mostly because of time constraints during development.

With the new servicing model that Microsoft introduced with Exchange 2013, cumulative updates (CUs) are released on a quarterly basis. A CU contains hotfixes, of course, but each CU also introduces new functionality. A CU is also a full package of Exchange 2013, so there’s no longer a need to install Exchange 2013 RTM followed by the subsequent updates, as was common practice in earlier versions of Exchange Server.

After releasing three CUs in a row, Microsoft released CU4 on February 25, 2014, which was equivalent to Exchange 2013 SP1. Is Exchange 2013 SP1 a new major release? The answer is yes and no. Exchange 2013 gradually evolved into SP1, but Microsoft has also released a lot of new features in SP1. An example of one of these features is the Edge Transport server role. This was available in Exchange 2007 and Exchange 2010, but not in Exchange 2013 RTM. The Edge Transport role has returned in Exchange 2013 SP1. Another example is support for SSL Offloading, which was available in Exchange 2010 but not in Exchange 2013 RTM. This is an example of that time constraint: during development of Exchange 2013 RTM there wasn’t enough time to test this feature properly and thus support it sufficiently.

Looking at the new servicing model with the cumulative updates you might ask why Microsoft is calling CU4 “SP1” instead of continuing the CU numbering. A new version, such as SP1, is a major milestone in a product’s lifecycle and as such is supportable. Exchange 2013 RTM will be supported for 10 years, and Exchange 2013 SP1 will be supported for 12 months after the release of the next Service Pack.¹ When it comes to supporting Cumulative Updates, Microsoft only supports the current CU version and the previous CU version. So, it’s purely a matter of supportability. The quarterly releases of CUs will continue, starting with the release of CU5 in the first quarter after release of SP1.

At first glance, Exchange 2013 RTM didn’t seem like a revolutionary change, but there was more than met the eye. Exchange 2013 is the first version from Microsoft to be designed from the ground up with the cloud in mind—in particular, Office 365. This is an area where Microsoft is facing tough competition from others, such as Google. Google Mail and Google Apps have a slick underlying infrastructure that makes it possible for users to add new features quickly and have good performance figures at the same time. This ability was something that hasn’t been Microsoft’s strongest point in the last couple of years, and therefore Microsoft decided to invest heavily in its cloud infrastructure. At the same time, Exchange Server was being redesigned to take advantage of these cloud developments.

¹The *Microsoft Product Lifecycle for Exchange 2013* states: “Support ends 12 months after the next service pack releases or at the end of the product’s support lifecycle, whichever comes first. For more information, please see the service pack policy at <http://support.microsoft.com/lifecycle/#ServicePackSupport>.”

What’s important in a public cloud environment like Office 365? It’s the scalability, architecture, and manageability of the platform. You’ll see these in the new front- and back-end architecture, in which the front end is the Client Access server acting as a protocol proxy. This is important in an environment with multiple data centers, perhaps in combination with a geographically dispersed DNS solution, in which you want your application to run with as few administrators and as little administrator input as possible. A solid monitoring situation with predefined actions and solutions is key to achieving such an environment.

Look at the JBOD (just a bunch of disks) solutions that Microsoft has been promoting since its introduction of Exchange Server 2010. This is a development driven by a need to lower the price of storage per GB. Running multiple copies of a mailbox database on simple SATA disks is easy to manage and has low replacement cost. That is, when a disk fails, which is not uncommon with cheap SATA disks, the Exchange server automatically moves over to another mailbox database on another disk. Exchange 2013 has the ability to automatically create mailbox database copies when spare disks are available, a feature called *auto reseal*. Later on, it’s a simple matter to rip and replace the faulty disk, and you’re back in business. This both decreases the cost of maintaining the disk infrastructure and lowers the operational cost of administrative staff.

Manageability is also an important factor when running a huge infrastructure in a data center. You don’t want to see an alert in your management console for every minor issue in your Exchange 2013 environment. This is where Managed Availability comes in; it will continuously monitor your Exchange 2013 environment and take appropriate action when needed. This action can include restarting an application pool in IIS, taking a process or service offline, or even rebooting a server. You can see this as a “self-healing” feature of Exchange 2013.

These are just a few key features for Microsoft data centers running Office 365, and you’ll see these features in the new Exchange 2013 as well. Does this mean that Exchange 2013 is targeted to large, multinational organizations? Well, yes and no. It’s yes in the sense that large, multinational organizations will certainly benefit from the new architecture with its front- and back-end technologies. But smaller organizations, perhaps with data-center resiliency, will certainly also benefit from Exchange 2013.

Larger organizations can create a combination of Exchange 2013 on premises and Office 365. This is called a “hybrid environment,” where the two are tightly integrated. Together they form one namespace with one address book, and yet they are independent where the actual mailboxes are located. Also, email sent between Exchange 2013 on premises and Office 365 is fully secure because of the configuration changes made by the hybrid configuration wizard.

This book is just like the product it describes. Originally it was written for Exchange 2013 RTM, but then it was updated for Exchange 2013 SP1, with additional content. A lot of the material applies to both versions, but when something only applies to SP1, it is noted as such.

Getting Started

To begin, let’s take a general look at Exchange 2013. First, we’ll consider the two Exchange 2013 editions and review their features. Then, we’ll look at their features compared to Exchange Server 2010, noting in particular which features are not part of Exchange 2013.

The Editions

Exchange 2013 is available in two editions:

- **Exchange 2013, Standard Edition.** This is a “normal” Exchange 2013, limited to five (5) mailbox databases per Mailbox server. This edition can also be used for Client Access servers.
- **Exchange 2013, Enterprise Edition.** This version can host up to 100 mailbox databases per Mailbox server. In Exchange 2013 RTM, the number of mailbox databases was limited to 50, but this was increased to 100 with the release of Exchange 2013 CU2. Just like the Standard Edition, this version can be used for Client Access servers. Considering the additional cost of an Enterprise Edition, it doesn’t make sense to use it for a single Exchange 2013 Client Access server.

Except for the number of mailbox databases per Exchange server, there are no differences between the two versions; the binaries are the same.

Entering the Exchange 2013 license key changes the limit of maximum mailbox databases for that server. Besides the Exchange 2013 server license, there's also a Client Access license (CAL), which is required for each user or device accessing the server software.

There are two types of CALs available:

- **Standard CAL.** This CAL offers standard email functionality from any platform. The license is for typical Exchange and Outlook usage.
- **Enterprise CAL.** This more advanced CAL offers functionality such as integrated archiving, compliance features, and information-protection capabilities. The CAL is an add-on to the Standard CAL, so both licenses need to be purchased!

This is not a complete list of all available features for the different CALs. For a complete overview, visit the Microsoft licensing page at <http://bit.ly/exlicense>.

What's New in Exchange 2013 SP1?

So, what are the new features and improvements in Exchange 2013? There are a lot of new features, valuable both from an administrator's point of view and from that of an end-user. In Exchange 2013 SP1, a new set of features is introduced as well, but let's discuss the most important changes here:

- **Support for Windows Server 2012 R2.** Long awaited in the Exchange community, Exchange 2013 SP1 now supports Windows Server 2012 R2. Please note that this only applies to Exchange 2013 SP1; unfortunately, Exchange 2013 RTM up to CU3 does *not* support Windows Server 2012 R2.
- **Edge Transport server.** This server role was available in both Exchange 2007 and Exchange 2010, but not in Exchange 2013 RTM. Thankfully the Edge Transport server role is again included in Exchange 2013 SP1, allowing companies standardizing on Exchange Server 2013 or Windows Server 2012 R2 to utilize the desired platform or product version. The Edge Transport server role functionality is "message hygiene," which means its primary purpose is anti-spam and anti-virus. Related to this is Exchange Online Protection, which is Microsoft's similar solution in the cloud. Both the Edge Transport server role and Exchange Online Protection are explained in detail in Chapter 6.
- **SSL Offloading.** SSL Offloading is another feature that was supported in Exchange 2010 but not in Exchange 2013 RTM. It has now been returned and Exchange 2013 SP1 supports SSL Offloading.
- **A new look and feel for client interfaces.** Exchange 2013 has a new appearance and tone across all messaging clients. Outlook 2013 has a new interface based on the new Microsoft design language. It's not an overloaded amount of information but, rather, offers a consistent view of all information, is easy to find, and is a snap to use. This interface can also be found in the Outlook Web App (OWA), as shown in Figure 1-1, and it's obvious that the OWA team and the Outlook 2013 team have worked closely together. This new design can be seen on all kinds of devices, with all types of clients or browsers. Use Windows 8 with Outlook 2013, or Windows 7 with OWA, or Windows Phone 8 with the Outlook mobile mail client, and they all offer this consistent view and user experience.

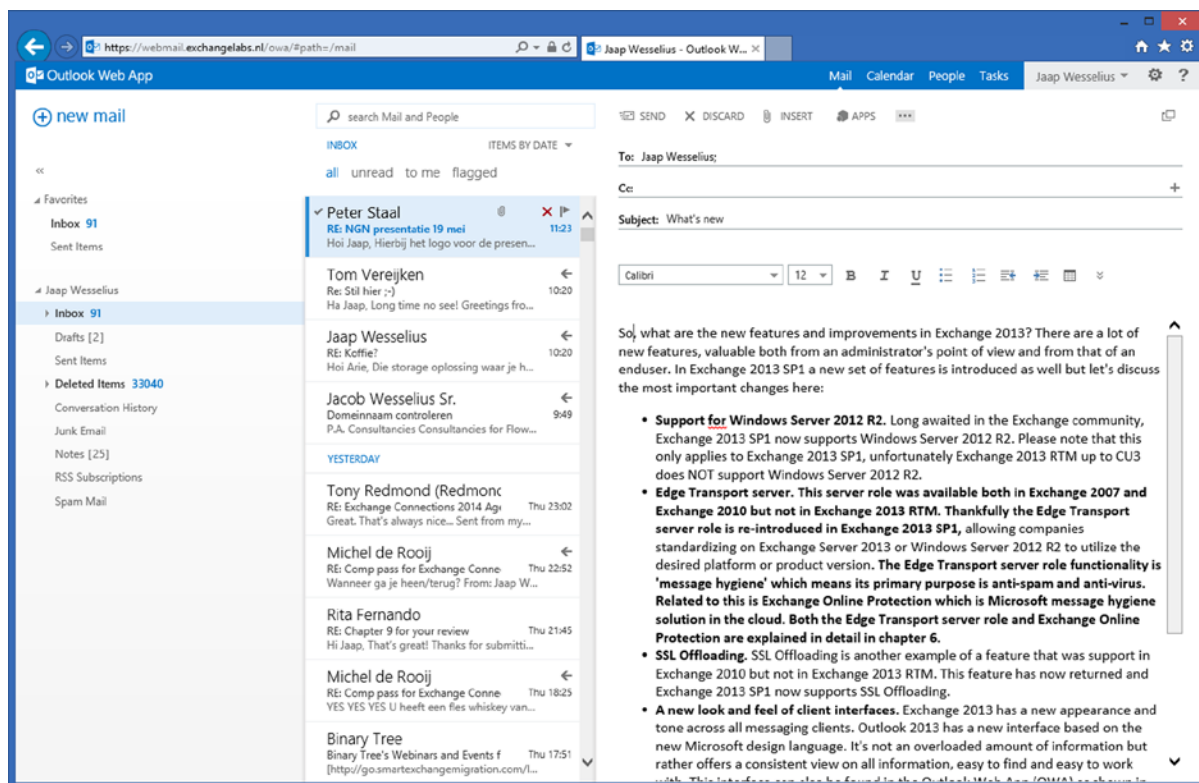


Figure 1-1. The new look and feel of Outlook Web App (OWA)

OWA also has a great new feature: When using Internet Explorer 10 or later (or Firefox 12, Safari 5.1, or Chrome 18, or later), you'll find OWA can be made available in offline mode, thereby giving you the option of working with OWA in an airplane, for example. Not all information is cached within the browser; it is comparable to a mobile client's use of ActiveSync, where only a few days' worth of data is stored. Only the default settings differ between ActiveSync and OWA offline.

Exchange 2013 SP1 has added three features to OWA:

- A rich text editor that make it possible to change the layout of newly created messages—for example, with other fonts, or to use bold or italic.
- An S/MIME control for Internet Explorer, making it possible to use secure messaging through S/MIME in OWA.
- DLP Policy tips.
- **Exchange Admin Center.** The Exchange Admin Center (EAC) is the new web-based management interface for Exchange 2013 (see Figure 1-2). Built on the new design for mail clients, it offers a management interface across various types of clients and web browsers and integration with Office 365 management.

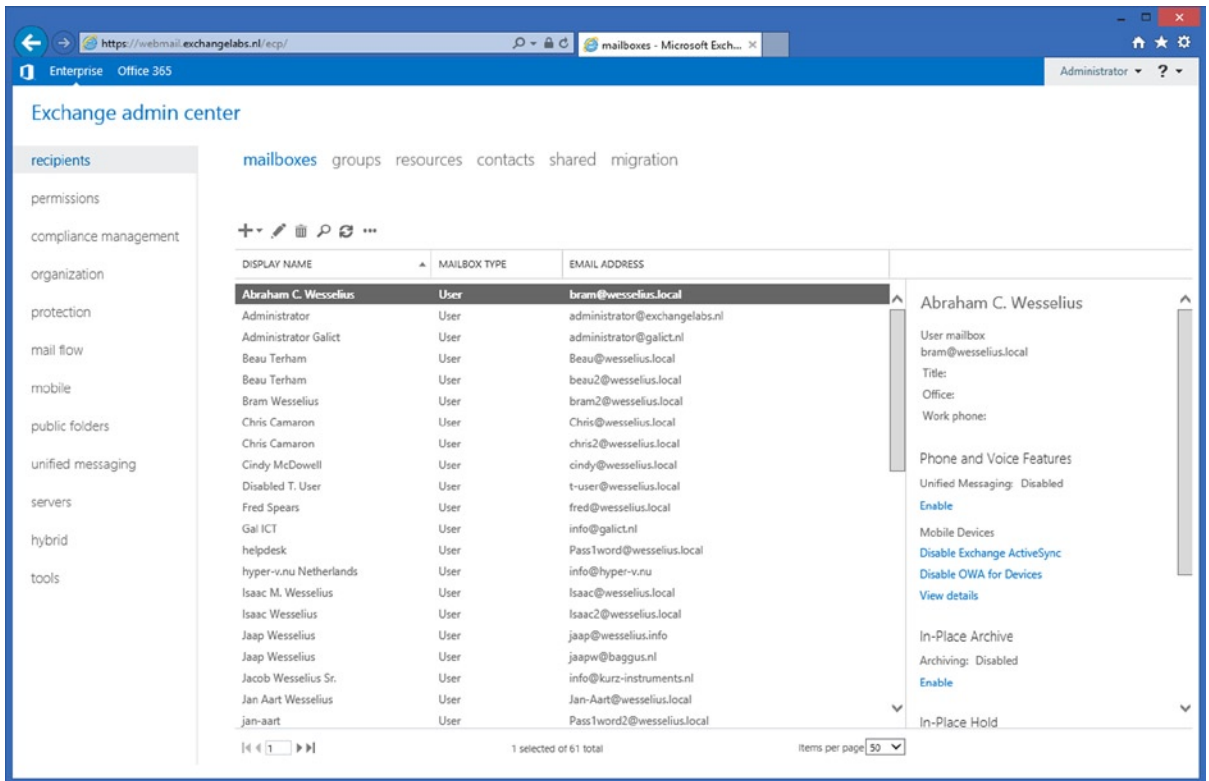


Figure 1-2. The new Exchange Admin Center (EAC) in Exchange 2013

Under the hood, EAC is using role-based access control (RBAC) so that only the management options enforced by RBAC are visible to the administrator. Just like the Exchange Management Console in Exchange Server 2010, not all the nitty-gritty details are available in the EAC; only the basic management functions are present. For all other management functions, the Exchange Management Shell (EMS) is available.

Exchange 2013 SP1 offers a new feature in EAC: cmdlet logging. That is, each action that's performed using EAC is translated under the hood in an Exchange Management Shell command, and this command is now shown in the cmdlet logging option. Yes, this feature was also available in Exchange 2007 and Exchange 2010, but it was *not* available in Exchange 2013 RTM.

The cmdlet logging is an extremely interesting feature for Exchange administrators because it allows us to learn PowerShell in a quick and easy way. Let's face it, as Exchange administrators we've been used to the GUI for so many years, but when we want to learn PowerShell and take advantage of its strength, we look for the easiest way to learn it. And that's using a cheat sheet like this.

- **Exchange Management Shell.** It's not really new in Exchange 2013, but the Exchange Management Shell (EMS) is strongly enhanced in this version. It now runs on top of PowerShell 3.0 (by default, in Windows Server 2012), with approximately 300 new cmdlets making it a powerful management tool. "EMS" and "PowerShell" are both used as names in this book, and part of its title is "PowerShell Administration," but when the name "PowerShell" is used in the text, we automatically mean the EMS.
- **Exchange 2013 architecture.** There's a new architecture when it comes to server roles. Only two server roles in Exchange 2013 RTM and three server roles in Exchange 2013 SP1, sometimes referred to as "building blocks," are available:
 - *Mailbox server role:* The Mailbox server role is the Exchange 2013 running in the back end, where all the mailboxes are stored. At the same time, the Mailbox server role contains the Hub Transport service and the unified messaging components. The Mailbox server role contains all the application logic needed to host a mailbox.
 - *Client Access server role:* The Client Access server role is running in the front end and is the server all clients connect to. It is responsible for authenticating the connection requests and proxies (or redirects, in case of SIP traffic) the requests to the appropriate Mailbox server. The Client Access server also contains the Front End Transport service (FETS) and a Unified Messaging (UM) Call Router.
 - *Edge Transport server role:* The Edge Transport server role is situated between the internal Exchange 2013 environment and the Internet, and it acts as an SMTP gateway. Typically, the Edge Transport servers are running in your network's demilitarized zone (DMZ), and as such they are not a member of your internal Active Directory environment. They commonly are workgroup members. The Edge Transport servers get their information via a synchronization mechanism called *edge synchronization*.
- **Managed store.** The "store" is the process running on the Exchange Mailbox server that's responsible for processing the mail transactions and storing the transactions in the mailbox databases. In Exchange 2013, the store process is completely rewritten in "managed code"—that is, C#, using the .NET Framework. More important, every mailbox database now has its own store process. So, even if one store process stops working, resulting in that particular mailbox database (copy) to stop working, the other mailbox databases on the same Mailbox server are unaffected. Earlier, in Exchange 2010, there was only one store process on a Mailbox server. When problems arose with the store process, all those mailbox databases were affected. This managed store is a great improvement in system stability.
- **Managed Availability.** One of the best new features of Exchange 2013 is its Managed Availability. It looks like some sort of "self-healing" feature, and it is responsible for monitoring all critical services on Exchange 2013. When needed, it takes appropriate action. Managed Availability consists of probes, monitors, and actions. *Probes* are constantly checking for certain services, and they feed the results into the monitors. The *monitors* evaluate the results from the probes. When needed, Managed Availability can perform certain *actions*. For example, it can check if OWA is up and running; if it's not, it can start or recycle the application pool where OWA is running or reset the Internet Information Services (IISRESET). Similarly, Managed Availability has probes for mailbox databases; if a mailbox database is found to be corrupted, Managed Availability can take action to automatically fail-over that mailbox database to another Mailbox server in the DAG and perform an automatic reseed of the corrupted mailbox database. This way, problems can be resolved even before end-users notice the failures, thereby reducing the number of calls to the help desk.

- **Outlook Anywhere.** This feature is not really new, but what's new in the Exchange 2013 environment is the fact that Outlook clients no longer connect using RPC over TCP (the traditional MAPI way). All Outlook clients now use RCP over HTTPS (i.e., Outlook Anywhere, or OA). This is true for both internal and external clients. So even an internal Outlook client automatically connects to the Exchange 2013 Client Access server (CAS) using RPC/HTTPS. The Outlook client is authenticated on the Exchange 2013 CAS, and after authentication, the request is proxied (again using RPC/HTTPS) to the Mailbox server where the mailbox is located.
- **MapiHttp.** Mapi over HTTP, codename Alchemy, is a new protocol in Exchange 2013 SP1, based on HTTP and positioned as a replacement for the RPC over HTTPS protocol. The idea behind this protocol is to remove the dependency on Remote Procedure Calls (RPC) when Outlook is communicating with the Exchange 2013 SP1 server. MapiHttp is only running on Exchange 2013 SP1 or later, and initially only Outlook 2013 SP1 will support it. Outlook 2010 will start supporting MapiHttp in a future update.
- **Anti-malware protection.** Exchange 2013 has built-in anti-malware protection available, but unfortunately it is not as feature-rich as the former Forefront Protection for Exchange (FPE), nor does it have the features that were available in the Exchange Server 2010 Edge Transport server. The anti-malware in Exchange 2013 is running one engine, and it scans messages that enter or leave the Exchange organization. If malware is found, it can remove the entire message or strip only the attachment if the malware is just in the attachment. For anti-spam and anti-virus solutions for SMTP in transit, Microsoft relies heavily on Exchange Online Protection (EOP), the successor to Forefront Online Protection for Exchange (FOPE), Microsoft's cloud solution for anti-spam and anti-virus. The good news is that both the Exchange Server 2010 and the Exchange Server 2007 Edge Transport server are running fine and are fully supported in combination with Exchange 2013, including Edge synchronization. For this to work correctly, though, you need Exchange 2007 SP3 RU10 or Exchange Server 2010 SP3.
- **"Modern" public folders.** Microsoft has invested heavily in public folders after years of uncertainty. They are calling the new version "modern public folders." The traditional public folders database has been discontinued in Exchange 2013, and it has been moved to the mailbox database. Because of this, the public folders are now protected by means of the database availability group (DAG) so that multiple copies of the public folders can exist in a DAG. Public folders themselves consist of the hierarchy (i.e., the folder structure) and the actual content. A writeable copy of the hierarchy is stored in a primary hierarchy mailbox, and there's only one writeable copy. The public folder content is stored in secondary hierarchy mailboxes; this is a new type of mailbox introduced in Exchange 2013. Besides public folder content, the secondary hierarchy mailboxes also contain a read-only copy of the hierarchy. Although public folders have migrated into these special mailboxes, Outlook clients and Outlook show them as "normal" public folders. Therefore, users will not notice the difference between the traditional public folders and the modern public folders.
- **Site mailboxes.** Site mailboxes are another new mailbox type in Exchange 2013, and they are a combination of Exchange 2013 and SharePoint Server 2013. That is, site mailboxes are designed for (temporary) project teams, where lots of Office documents are sent back and forth among members of the groups. Under the hood, these site mailboxes are actually a SharePoint team site that is much more capable of storing document-type information. For an Outlook client, it is fully transparent and the site mailbox is visible as a normal mailbox. This is a great example of "Exchange and SharePoint: Better Together."

- **Data Loss Prevention.** Data Loss Prevention, or DLP, is a new security feature in Exchange 2013. It's designed to prevent sending out messages that contain confidential information, based on Transport rules. For example, DLP can be used to filter messages that contain credit card numbers or Social Security numbers. It does this by checking the messages as they are submitted against certain predefined templates. If there's a match, a warning is displayed—much like mail tips—about what DLP has found to be a security issue. This is fully configurable so it can match your security requirements. A number of predefined DLP policies are included in Exchange 2013, and the policies are customizable to fit company policies.

Of course, there are many more new features in Exchange 2013, but these are the most important ones.

What Has Been Removed from Exchange Server

Every new version of Exchange Server introduces new features, but at the same time other features are discontinued, deprecated, or available only in some other form or scenario. The most important changes or discontinued features in Exchange 2013 are:

- **Support for Outlook 2003.** Outlook 2003 is not supported in Exchange 2013. Not only is it unsupported, it is not working. Outlook 2003 depends on system folders, free/busy, and offline address book distribution folders in public folders, and these system folders have been discontinued. Unfortunately, there is still a huge installed base for Outlook 2003, so this could be a major showstopper in the deployment of Exchange 2013.
- **RPC/TCP access for Outlook clients.** The traditional RPC/TCP access for Outlook clients is no longer supported in Exchange 2013. All Outlook clients will connect using Outlook Anywhere (OA, formerly known as RPC/HTTPS), whether they are on the internal or the external network. The reason is obvious; RPC/HTTPS is easier to route via the networks because it requires only port 443 to be open on firewalls. For RPC/TCP this is not the case and most firewalls block RPC traffic.
- **Hub Transport Server role.** The dedicated Hub Transport server that was used in Exchange Server 2007 and Exchange Server 2010 is no longer available as a dedicated server. Instead, it is integrated into the Mailbox server role, so that every Mailbox server automatically has a transport service installed. This transport service is responsible for routing SMTP messages, both inside the Exchange Service organization and to the Internet. The Exchange 2013 CAS is a protocol proxy for the transport service on the Mailbox server; the service on the Exchange 2013 CAS is called Front End Transport (FET). External SMTP hosts connect to the FET on the Exchange 2013 Client Access server, which proxies the request to the transport service running on the Mailbox server where the recipient's mailbox is located. The Mailbox server can route SMTP messages directly to the Internet, but it can also use the Exchange 2013 CAS as a front-end proxy for outbound messages.
- **Unified Messaging server role.** The dedicated Unified Messaging (UM) server role is no longer available as a dedicated server. Just like the Hub Transport server, it is now integrated with the Exchange 2013 Mailbox server. When you are installing an Exchange 2013 Mailbox server, the UM service is automatically installed. For SIP traffic, the Exchange 2013 CAS does not act as a proxy, but the UM Call Router service redirects the SIP request to the UM service on the Mailbox server where the recipient's mailbox is located.

- **Exchange Management Console and Exchange Control Panel.** In previous versions of Exchange Server, the Exchange Management Console (EMC) was the primary graphical UI for managing the entire Exchange environment. While this worked fine in a smaller environment, it was less usable in large, multi-center environments. In Exchange 2013, Microsoft has discontinued the EMC and it is replaced by the Exchange Admin Center (EAC). The same is true for the Exchange Control Panel (ECP). It has been discontinued in Exchange 2013, and user self-management is now performed by the EAC.
- **Managed folders.** Managed folders were introduced in Exchange Server 2007 as Microsoft's solution for information management and compliance. In Exchange Server 2010, Microsoft introduced the personal archive and retention policies; as a result, managed folders in Exchange Service 2010 were deprecated. This was clearly visible in Exchange Server 2010 SP1, where managed folders were manageable only from the EMS and were not compatible with the personal archive. In Exchange 2013, managed folders have been decommissioned.
- **Anti-spam agent management.** Anti-spam functionality as we knew it in Exchange Server 2010 is not available in Exchange 2013. The Exchange Service 2013 CAS does not perform any anti-spam duties, so all SMTP messages are proxied to the transport service on the Exchange 2013 Mailbox servers. These do have some anti-spam functionality, but compared to the Exchange Server 2010 Hub Transport server or the Exchange 2013 SP1 Edge Transport server, they are very limited.
- **Forefront Protection for Exchange.** The anti-malware built into Exchange 2013 is limited and absolutely not comparable to Microsoft's Forefront Protection for Exchange (FPE), which was previously available. Now, anti-malware is available only on the Mailbox server in the back end. There are no options for managing the anti-malware solution other than to turn it on or off or to change the notification text.

A bit beyond the scope of this book is the Forefront Threat Management Gateway (TMG) 2010. At the end of 2012, Microsoft announced the end of life for TMG 2010. While TMG will be supported for a couple of years, it will continue to work with Exchange Server 2010—and with some minor adjustments, it will also work with Exchange 2013. For the long term, however, it is recommended you start looking for alternatives to this firewall and pre-authentication functionality. Right after the end-of-life announcement, the official Microsoft alternative was to use its Forefront Unified Application Gateway (UAG), but then Microsoft announced the end of its life as well.

When publishing Microsoft services toward the Internet, using some sort of reverse proxy, you have to look for other alternatives. When it comes to reverse proxy, the Application Request Routing (ARR) module is available for running on top of the Internet Information server (IIS) or the Web Application Proxy (WAP) in Windows Server 2012 R2, but other third-party hardware vendors (like Cisco, Juniper, Kemp, or F5) can deliver the same functionality, sometimes combined with load-balancing functionality.

Integration with Active Directory

Active Directory is the foundation for Exchange 2013, as it has been for Exchange Server since Exchange 2000 was released 14 years ago. Earlier versions of Exchange Server—that is, Exchange 5.5 and earlier—relied on their own directory, which was separate from the (NT4) user directory. Active Directory stores most of Exchange's configuration information, both for server/organization configuration and for mail-enabled objects.

A Microsoft Windows Active Directory Directory Service (ADDS) is best described as a forest; this is the highest level in the Directory Service and is the actual security boundary. The forest contains one or more Active Directory Directory domains; a domain is a logical grouping of resources, such as users, groups, and computers. An Exchange 2013 organization is bound to one forest, so even if you have an environment with over 100 domains, there can be only one Exchange organization.

Active Directory sites also play an important role in Exchange deployment. An Active Directory site can be seen as a location, well connected with high bandwidth and low latency—for example, a data center or an office. Active Directory sites can contain multiple Active Directory domains, but an Active Directory domain can also span multiple Active Directory sites.

Exchange 2013 depends heavily on ADDS, and these need to be healthy. The minimum levels in ADDS need to be Windows 2003 Forest Functional Level (FFL) and Windows 2003 Domain Functional Level (DFL). The Domain Controllers need to be at a minimum level of Windows Server 2003 SP1.

Active Directory Partitions

A Microsoft Windows ADDS consists of three system-provided partitions:

- **Schema partition.** The schema partition is the blueprint for all objects and properties that are available in Active Directory. For example, if a new user is created, a user object is instantiated from the schema, the required properties are populated, and the user account is stored in the Active Directory database. All objects and properties are in the schema partition, and therefore it depends which version is used. Windows 2012 R2 Active Directory has much newer objects and newer (and more) properties than, for example, Windows 2003 Active Directory. The same is true, of course, for applications like Exchange Server. Exchange 2013 adds a lot of new objects and attributes to Active Directory that make it possible to increase functionality. Therefore, every new version of Exchange Server, or even the cumulative updates or service packs, needs to make schema changes.

There is only one schema partition in the entire Active Directory forest. Even if you have an Active Directory forest with 100 domains and 250 sites worldwide, there's only one schema partition. This partition is replicated among all Domain Controllers in the entire Active Directory forest. The most important copy of the schema partition is running on the schema master, which is typically the first Domain Controller installed in the forest. This copy is the only read-write copy in the entire Active Directory forest.

- **Configuration partition.** The configuration partition is where all nonschema information is stored that needs to be available throughout the Active Directory forest. Information that can be found in the configuration partition is, for example, about Active Directory sites, about public key infrastructure, about the various partitions that are available in Active Directory, and of course about Exchange Server. Just like the schema partition, there's only one configuration partition. It replicates among all Domain Controllers in the entire Active Directory environment so that all the Exchange servers have access to the same, consistent set of information. All information regarding the Exchange server configuration, like the Exchange servers themselves, the routing infrastructure, or the number of domains that Exchange Server is responsible for, is stored in the configuration partition.
- **Domain Partition.** The domain partition is where all domain-specific information is stored. There's one partition per domain, so if you have 100 domains in your Active Directory forest, you have 100 separate domain partitions. User objects, contacts, and security and distribution groups are stored in the domain partition.

The best tool for viewing the three Active Directory partitions is the ADSI Edit MMC (Microsoft Management Console) snap-in, which is shown in Figure 1-3.

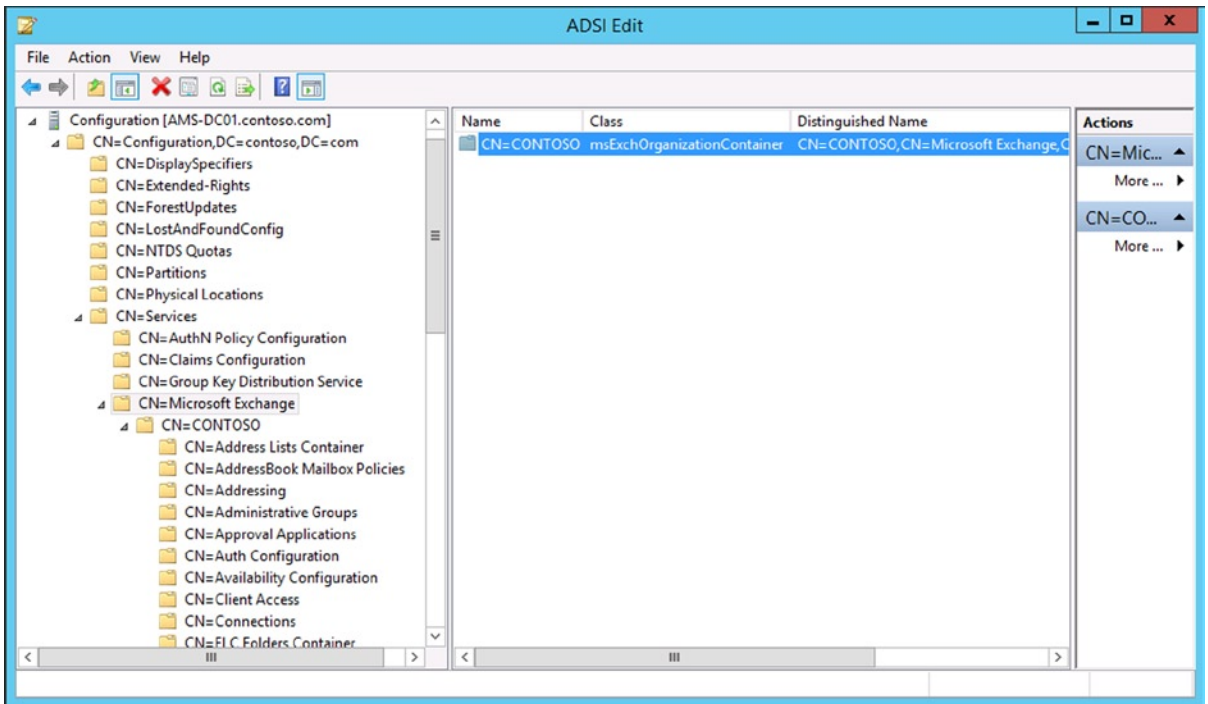


Figure 1-3. The Exchange information is stored in the configuration partition

■ **Warning** There's very little safeguarding in this tool, so it's easy to destroy critical parts in Active Directory when you're just clicking around!

The Active Directory Users and Computers (ADUC) MMC has a focus on the domain partition. In Windows Server 2012, the Active Directory Administrative Center (ADAC) is the preferred tool to manage the Active Directory environment. But using either tool is relatively safe, since the tool prevents messing around with objects in a way that Active Directory does not like. The Active Directory Sites and Services (ADSS) MMC snap-in reads and writes information from the configuration partition. All changes made here are visible to all domains in the forest; the same is true for the Active Directory domains and trusts MMC snap-in.

A very powerful tool regarding Active Directory is the Schema MMC snap-in, which is usually run on the Domain Controller that holds the schema master role. Using the Schema MMC snap-in, it is possible to make changes to the Active Directory schema partition.

■ **Warning** Only do this when you're absolutely sure of what you're doing, and when you have proper guidance—for example, from Microsoft support. Changes made to Active Directory can be irreversible!

Domain Controllers also have tools like LDIFDE and CSVDE installed as part of the AD management tools. These are command-line tools that can be used to import and export objects into or out of Active Directory. LDIFDE can also be used to make changes to the Active Directory schema, and the Exchange 2013 setup application uses the LDIFDE tool to configure Active Directory for use with Exchange 2013. These tools are beyond the scope of this book.

When promoting a server to a Domain Controller, or when installing the Remote Server Administration Tools (RSAT) for Active Directory Directory Services (ADDS), the PowerShell Active Directory module is installed as well. This module enables Active Directory functionality in PowerShell, making it possible to manage Active Directory using PowerShell cmdlets.

Active Directory Permissions

There are three partitions in Active Directory. Each of these partitions has separate permissions requirements, and not everybody has (full) access to these partitions. The following are the default administrator accounts or security groups that have access to each partition.

- **Schema Admins security group.** The Schema Admins have full access to the schema partition. The first administrator account is the top-level domain, which is the first domain created. To make the necessary changes to the schema partition for installing Exchange Server, the account that's used needs to be a member of this security group. Any other domain administrator in the forest is, by default, not a member of this group.
- **Enterprise Admins security group.** The Enterprise Admins have full access to the configuration partition. Again, the first administrator account in the top-level domain is a member of this group and as such can make changes to the configuration partition. Since all Exchange Server configuration information is stored in the configuration partition, the account used for installing Exchange Server needs to be a member of this group. Please note that the Enterprise Admins security group does not have permission to make changes to the schema partition.
- **Domain Admins security group.** The Domain Admins have full access to the domain partition of the corresponding domain. If there are 60 domains in an Active Directory environment, there are 60 domain partitions and thus 60 different Domain Admins security groups. The first administrator account in the top-level domain is a member of the Domain Admins security group in this top-level domain.

Why is this important to know? In the early days of Active Directory, Microsoft recommended using multiple domains in an Active Directory forest, preferably with an *empty root* domain. This empty root domain is a domain without any resources, and its primary purpose was for Active Directory management. All resources like servers, computers, users, and groups were located in child domains. Needless to say, this has some implications for the use of various administrator accounts. It is a delegated model, where the administrator accounts in the top-level domain have control over all Active Directory domains, whereas the administrators in the other domains have administrative rights only in their respective Active Directory domains. These other administrators do not have administrative privileges in other domains, let alone permission to modify the configuration partition or the schema partition.

But things have changed, and although an empty root Active Directory domain environment can still be used, it is no longer actively recommended. Mostly recommended these days is a "single forest, single domain" environment unless there are strict legal requirements that dictate using another Active Directory model.

Chapter 10 will explain about security in great detail and will explore the various options available for delegated administration and split permissions. But in short, the default administrator account that's created in the top-level Active Directory domain has enough permissions for installing Exchange 2013.

Active Directory Sites

Active Directory sites play an important role in the larger Exchange 2013 deployments. As stated earlier, an Active Directory site can be seen as a (physical) location with good internal network connectivity, high bandwidth, and low latency—that is, a local LAN. An office or data center is typically a good candidate for an Active Directory site.

An organization can have multiple locations or multiple data centers, resulting in multiple Active Directory sites. Sites are typically interconnected, with lower bandwidth, higher latency connections. An Active Directory site can also have multiple domains, but at the same time an Active Directory domain can span multiple sites.

An Active Directory site also is a replication boundary. Domain Controllers in an Active Directory site replicate their information almost immediately among Domain Controllers in the same site. If a new object is created, or if an object is changed, the other Domain Controllers in that same site are notified immediately and the information is replicated within seconds. All Domain Controllers in an Active Directory site should contain the same information.

Information exchanged between Domain Controllers in different Active Directory sites is replicated on a timed schedule, defined by the administrator. A typical timeframe can be 15 minutes, but depending on the type of connection or the bandwidth used to a particular location (you don't want your replication traffic to interfere with normal production bandwidth), it can take up to several hours. This means that when changes are made to Active Directory—for example, when installing Exchange 2013—it can take a serious amount of time before all the information is replicated across all the Domain Controllers and the new changes are visible to the entire organization.

Active Directory sites are created using the Active Directory Sites and Services MMC snap-in (see Figure 1-4). The first step is to define the network subnets in the various locations in the snap-in, and then tie the actual Active Directory site to the network subnet. For example, a data center in the Amsterdam site has IP subnet 10.38.96.0/24, while the data center in the London site has IP subnet 10.38.97.0/24.

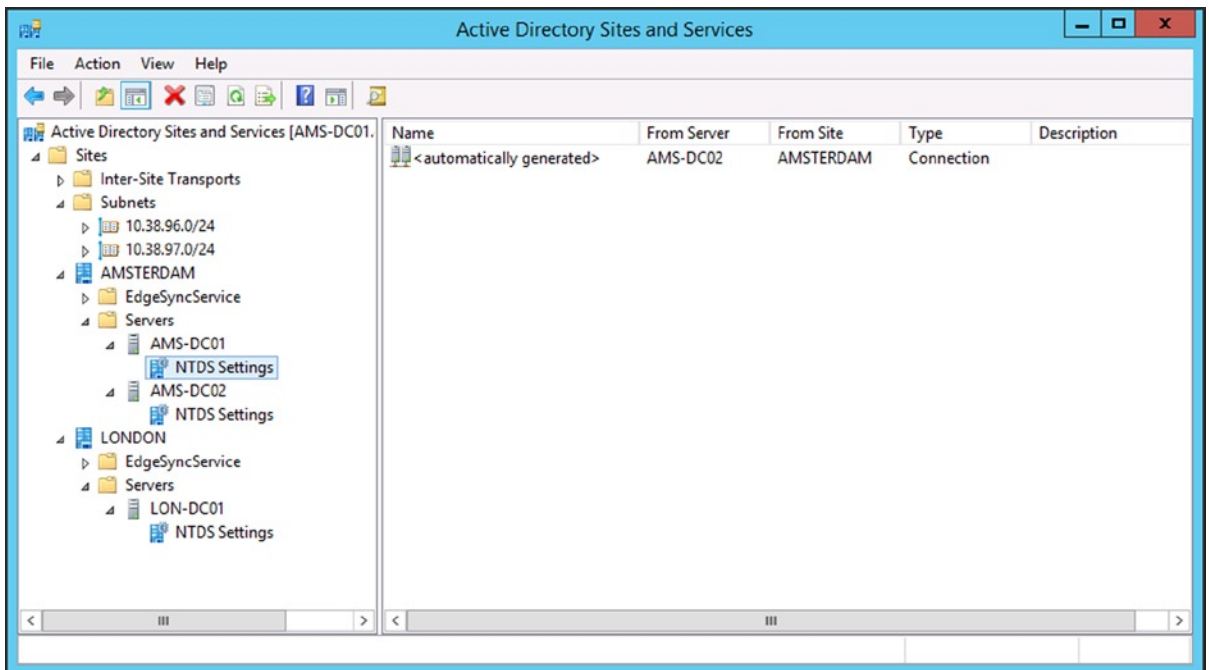


Figure 1-4. Two different subnets and sites, as shown in Active Directory Sites and Services

A location like a data center in London or in Amsterdam (which corresponds with the Active Directory sites) can be “Internet facing” or “non-Internet facing,” a descriptor that indicates whether the location has Internet connectivity or not. This is important for Exchange 2013, since it determines how namespaces are configured and thus how external clients are connected to their mailboxes in the different locations.

For example, the environment in Figure 1-4 has two Active Directory sites. If the data center in Amsterdam has an Internet connection and the data center in London does not, all clients from the Internet are connected initially to the Exchange 2013 servers in Amsterdam. If a user’s mailbox is located in London, the client request is proxied to the Exchange 2013 servers in London.

But, if the data center in London also has an Internet connection and the Exchange servers are configured accordingly, the London-based clients can access the Exchange 2013 servers from the Internet in Amsterdam, though the request will be redirected to the Exchange 2013 servers in London and thus connect directly to the servers in London.

Also, the routing of SMTP messages through the Exchange organization is partly based on Active Directory sites. In the example just given, it is not that difficult to do, but if you have an environment with dozens of Active Directory sites, the SMTP routing will follow the Active Directory site structure unless otherwise configured.

Exchange 2013 Architecture

Exchange 2013 is using what they have termed “building blocks”; there are three such building blocks:

- **Client Access server.** The Client Access server (CAS) is the server where all clients connect. The CAS consists of three parts: Client Access Front End (CAFE), Front End Transport Service (FETS), and the UM Call Router (UMCR). The CAS performs authentication of a client request, it locates the location of the client’s mailbox, and it proxies or redirects the client request to the appropriate Mailbox server, where the actual client mailbox is located. The CAS in Exchange 2013 is sometimes also referred to as the “front end,” although according to the book, UMCR is not officially a front end.
- **Mailbox server.** The Mailbox server is the server where the actual mailbox data is stored. Clients do not access the Mailbox server directly; all requests are routed through the CAS. The Mailbox server in Exchange 2013 is sometimes also referred to as the “back end.” Rendering for clients like OWA, transport transcoding for SMTP, or voice processing for the UM role *always* takes place on the Mailbox server.
- **Edge Transport server.** The Edge Transport server is used for message hygiene purposes and acts as an SMTP gateway between your internal Exchange environment and the Internet. When an Edge Transport server is used, all messages are routed through this server. Using an Edge Transport server is not mandatory; there are lots of customers who have decided not to use an Edge Transport server and use a third-party solution instead.

In Exchange Server 2007 and Exchange Server 2010, the Hub Transport server and the Unified Messaging server were also dedicated server roles, next to the Client Access server and the Mailbox server. These four server roles were tightly coupled and they used RPC for inter-server communication. Although this works fine, it presents some challenges when it comes to an environment with multiple data centers and to site resiliency. One of the design goals for Exchange 2013 was to remove the tight coupling of the server roles and replace them with a more loosely coupled mechanism.

Hence, the four server roles are no longer available in separate server roles. The Client Access server continues to exist as a dedicated server, but the other three server roles are incorporated into the Mailbox server role. When installing the latter, the Hub Transport and Unified Messaging services are automatically installed as well. The Mailbox server contains most of the business logic of Exchange 2013, and this is the server where all the processing takes place for all mailboxes located on that Mailbox server. And since all business logic and processing takes place on the Mailbox server, the Client Access server has a relatively light service role.

Microsoft has a nice poster, which is a large PDF ready for printing, showing the entire Exchange 2013 architecture; it is available at <http://bit.ly/ExPoster>.

The Client Access Server

The Client Access server (CAS) performs only authentication of a client request; after authentication, the client request is proxied to the Mailbox server where the destination mailbox is located. The CAS in itself does not perform any processing with respect to mail data. Compared to previous versions of Exchange Server, the Exchange 2013 CAS is basically a “thin” server. According to Microsoft, its connections are stateless (not clueless, though). But the connections are not really stateless because the SSL connection is terminated at the CAS and then processed. If a CAS goes offline, all connections are terminated and they have to be set up again on another CAS (which would not be the case in a true stateless setup). The reason that Microsoft calls it “stateless” is that there’s no persistent storage on Exchange 2013 CAS.

Unlike Exchange Server 2010 and Exchange Server 2007, the CAS no longer communicates with the Mailbox server using RPC; the original client request is instead proxied to the Mailbox server using the same protocol as was used when the connection reached the CAS server. If the initial request from the client to the Client Access server is from Outlook Web App (so HTTPS), the protocol between the CAS and the Mailbox server is also HTTPS. Note that the request from Internet to the CAS is using the regular port 443, but that the proxied request to the Mailbox server is using port 444.

The same is true for other protocols like POP3, IMAP4, and SMTP. After the Exchange 2010 Client Access server receives the request, the Client Access server proxies the request to the Exchange 2013 Mailbox server, as shown in Figure 1-5. An exception is the SIP protocol. When a SIP request is received from a Lync server, the Client Access server determines the appropriate Mailbox server, but instead of proxying the request, the Client Access server redirects the request to the appropriate Mailbox server. From this moment on the Lync server communicates directly with the Mailbox server. This is also clearly visible in Figure 1-5.

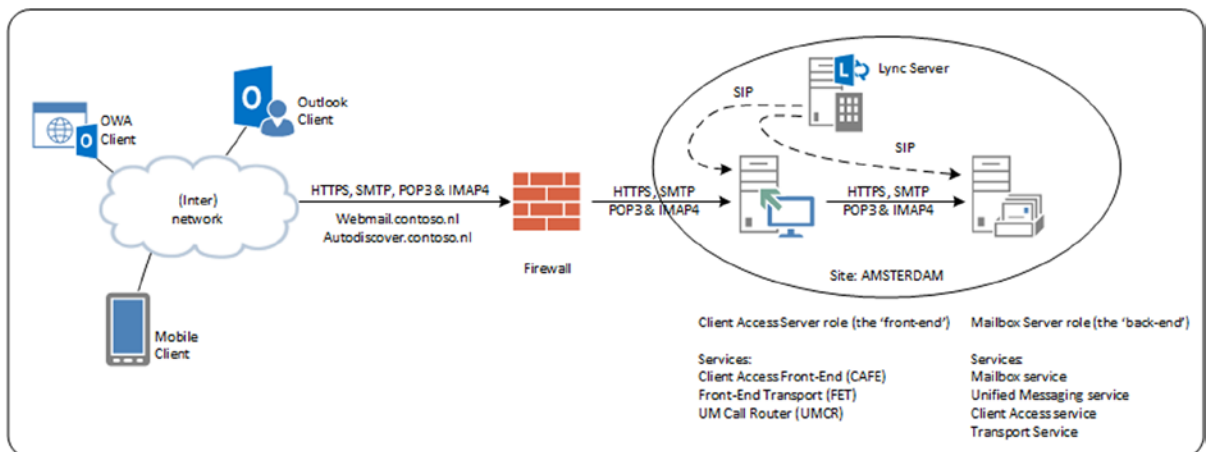


Figure 1-5. The front-end and back-end architecture in Exchange 2013

This architecture means that the actual Exchange 2013 servers are now loosely coupled, which offers huge advantages when multiple offices or multiple data centers are used.

As stated before, the CAS is a “thin” server and does not store any information from the sessions, except for the various protocol proxy logs like Autodiscover, Outlook Anywhere, or IIS logging. This is true for both regular client requests and SMTP requests. SMTP requests are accepted by the Client Access Front End service on the CAS, but the message itself is not stored or queued on the CAS, as it is on an Exchange 2010 Edge Transport server, for example.

The Front End Transport service that is responsible for handling SMTP messages on the CAS doesn’t store messages on the server itself, but passes the SMTP messages directly to the appropriate Mailbox server where the intended recipient’s mailbox is located, or to a down-level Hub Transport server if the recipient is located on a down-level Mailbox server. The Front End Transport service does not inspect message content.

Because of the stateless connections from clients, the load-balancing solution needed when multiple CAS are used doesn't have to be a layer 7 load balancer, as used to be the case in Exchange 2010; Exchange 2013 works fine with (much simpler) layer 4 load balancers.

The Mailbox Server

The Mailbox server is where all the processing regarding messages takes place. Clients connect to the CAS, but the requests are proxied or redirected to the appropriate Mailbox server or to another down-level CAS server. All message rendering takes place on the Mailbox server, in contrast to Exchange Server 2010, where all rendering took place on the CAS itself. To achieve this, there's also a CAS component on the Mailbox server.

SMTP Transport is now also located on the Mailbox server and consists of three separate services:

- The Transport service
- The Mailbox Transport Delivery service
- The Mailbox Transport Submission service

The Transport service can be seen as the successor to the "old" Hub Transport server, and it handles all SMTP message flow within the organization, such as routing, queuing, bifurcation, message categorization, and content inspection. Important to note is that the Transport service never communicates directly with the mailbox databases. Communication between the Transport service and the mailbox database is performed by the Mailbox Transport Delivery service and the Mailbox Transport Submission service. These services connect directly to the mailbox database (using RPC!) to deliver or retrieve messages from the mailbox database. As with the Front End Transport service, the Mailbox Transport Delivery and Mailbox Transport Submission services do not queue any messages on the Mailbox server; the Transport service (notice the absence of the word *mailbox*) does queue information on the Mailbox server. (The transport mechanism is covered in detail in Chapter 4.)

The most important part of this, of course, is the mailbox components that run on the Mailbox server. The information store, or store process, is responsible for handling all mailbox transactions and for storing these transactions in a mailbox database. The database is not a relational database like SQL server; it's running on its own engine, the extensible storage engine or ESE. The ESE database engine has been fully optimized for the past 15 years for use with Exchange Server, so it performs very well and is very reliable. The ESE database is a transactional database using a database, log files, and a checkpoint file. (I'll get back to database internals in Chapter 4.)

The Exchange Replication service is another important service running on the Mailbox server. This service is responsible for replicating mailbox data from one mailbox database on one Mailbox server to a mailbox database running on another Mailbox server. The collection of Mailbox servers replicating data between each other is called the Database Availability Group, or DAG. A DAG can take up to 16 Mailbox servers. Each mailbox database has one active mailbox database copy, and may have up to 15 passive mailbox database copies.

The database in Exchange 2013 has been greatly improved compared to earlier versions. For instance, Exchange 2013 now generates 50 percent fewer IOs per second (IOPS) compared with Exchange Server 2010, making it now possible to store multiple databases, including their log files, on one physical disk. This is something that Microsoft never recommended doing in the past, but now it is a viable solution. Of course, this is recommended only when there are multiple copies of a mailbox database available for redundancy purposes and after proper sizing to ensure that the disk will be able to handle the total number of IOPS.

Exchange 2013 Management

When it comes to Exchange 2013 Management, there are major changes compared to the previous versions of Exchange. There are two options for managing your Exchange 2013 environment:

- Exchange Admin Center – The HTML-based GUI that offers the most basic options for managing your Exchange 2013 environment
- Exchange Management Shell – The command-line interface running on top of Windows PowerShell and offering all nitty-gritty options when managing your Exchange 2013 environment

I'll discuss these in more detail, as follows.

Exchange Admin Center

The Exchange Admin Center (EAC) is the GUI for managing your Exchange 2013 environment. The EAC can be managed from the internal network as well as from the external network. The EAC is accessible via a URL like <https://webmail.contoso.com/ecp>, and when the EAC is opened, a window like the one shown in Figure 1-6 appears.

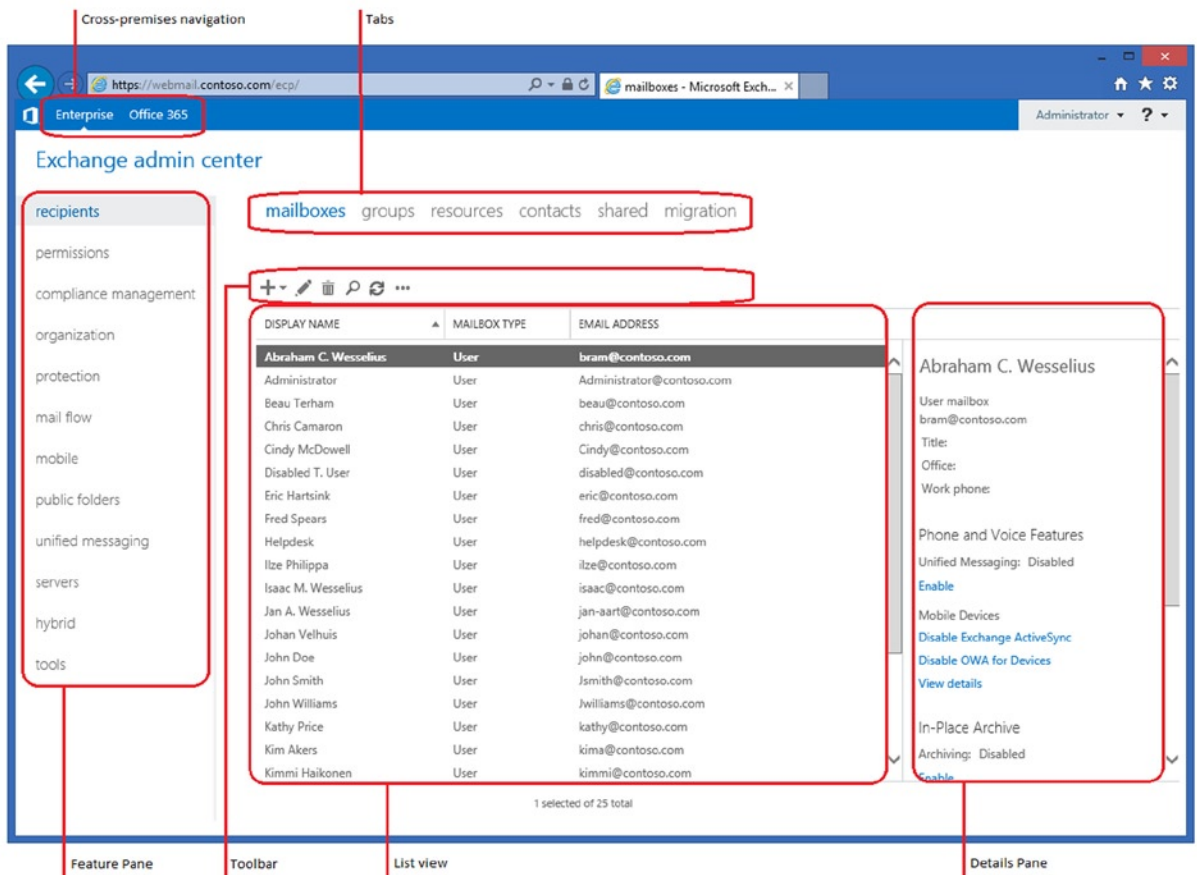


Figure 1-6. The new Exchange Admin Center in Exchange 2013 SP1

■ **Note** There were little changes with respect to EAC in Exchange 2013 SP1. The tools section in the Feature pane is new.

In the left-hand menu, there are various components of Exchange 2013 that can be managed in the EAC. The left-hand menu is also called the “Feature pane” and consists of the following features:

- **Recipients.** All recipients, like mailboxes, groups, contacts, shared mailboxes, and resource mailboxes, are managed from the Recipients option.
- **Permissions.** In the Permissions option, you can manage administrator roles, user roles, and Outlook Web App policies. The first two roles are explained in more detail in the RBAC section later in this chapter.
- **Compliance Management.** In the Compliance Management option, you can manage In-Place eDiscovery, In-Place Hold, auditing, data loss prevention (DLP), retention policies, retention tags, and journal rules.
- **Organization.** The Organization option is the highest level of configuration, and this is the place where you’ll manage your Exchange organization, including federated sharing, Outlook Apps, and address lists.
- **Protection.** In the Protection option, you can manage anti-malware protection for the Exchange 2013 organization.
- **Mail Flow.** The Mail Flow option contains all choices regarding the flow of messages, including transport rules, delivery reports, accepted domains, email address policies, and send and receive connectors.
- **Mobile.** All settings regarding mobile devices are managed from the Mobile option. You can manage mobile device access and mobile device mailbox policies.
- **Public Folders.** The Public Folder Management Console in Exchange Server 2010 is replaced by this feature in the EAC. From the Public Folders option you can manage Exchange 2013 public folders. Note that legacy public folders cannot be managed using the EAC.
- **Unified Messaging.** From the Unified Messaging option you can manage UM Dial Plans and UM IP Gateways.
- **Servers.** The Exchange 2013 servers, both Mailbox and Client Access server, can be managed from the Servers option. This also includes databases, database availability groups (DAGs), virtual directories, and certificates.
- **Hybrid.** Using the Hybrid option, it is possible to configure a hybrid organization—that is, you can connect your on-premises Exchange 2013 organization with an Office 365 tenant.
- **Tools.** This is new in Exchange 2013 SP1 and contains links to Exchange and Office 365 specific tools.

■ **Note** Just like the EMS, the functions available in the EAC are limited by the permissions enforced by RBAC.

In previous versions of Exchange Server, all the configuration changes made using the Exchange Management Console were translated to EMS under the hood. In the EAC, this is no different, and like the EMC, these commands are logged in a special section so you can check out the actual commands. Although small, this is one of the great improvements in Exchange 2013 SP1, as this feature was not available in Exchange 2013 RTM!

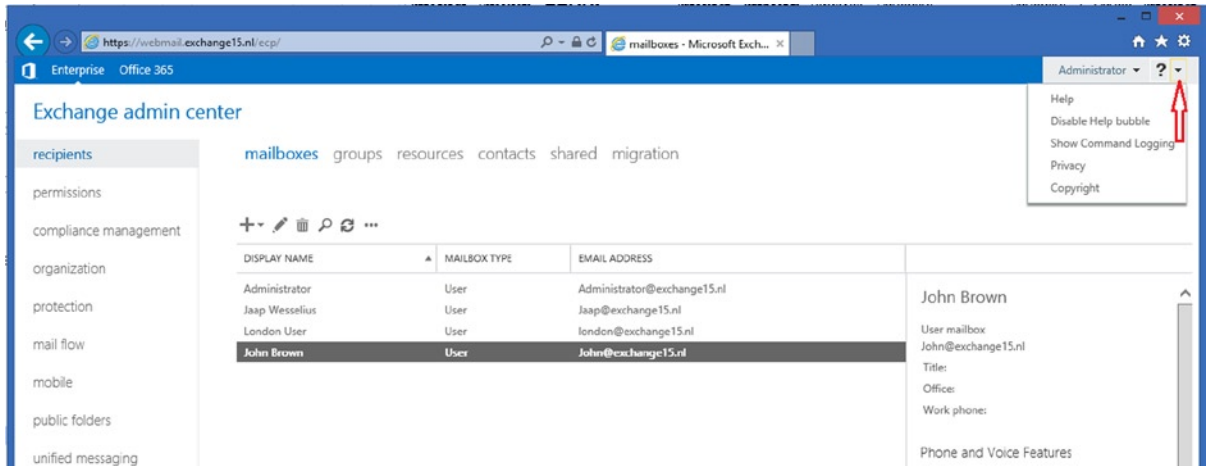











Figure 1-7. The command log is available in the upper right menu in EAC

The tabs in the top-level menu are context sensitive. In other words, they change when a different feature in the Feature pane is selected.

The toolbar can be compared to the Actions pane in the Exchange Server 2010 Management Console. All actions are associated with an icon. Table 1-1 describes each of the icons.

Table 1-1. Available Options (icons) in the EAC Toolbar

Icon	Name	Description
	Add, New	Use this option to create a new object. Some of these icons have an associated down arrow you can click to show additional objects you can create. For example, in Recipients and then Mailboxes, clicking the down arrow displays User Mailbox and Linked Mailbox as additional options.
	Edit	You can use this option to edit an object.
	Delete	Use this option to delete one or more objects.
	Search	Use this option to query for a particular object.
	Refresh	Use this icon to refresh the objects in the list view.
	More options	Use this icon to view more actions you can perform for that tab's objects. For example, in Recipients > Mailboxes, clicking this icon shows the following options: Disable, Add/Remove Columns, Export Data to a CSV File, Connect a Mailbox, and Advanced Search.
	Up and Down arrow	Use these icons to move an object's priority up or down. For example, in Mail Flow and then Email Address Policies, click the up arrow to raise the priority of an email address policy. You can also use these arrows to navigate the public folder hierarchy or to move rules up or down in the list view.
	Copy	Use this icon to copy an object so you can make changes to it without changing the original object. For example, in Permissions and then Admin Roles, select a role from the list view, and then click this icon to create a new role group based on an existing one.
	Remove	Use this icon to remove an item from a list. For example, in the Public Folder Permissions dialog box, you can remove users from the list of users allowed to access the public folder by selecting the user and clicking this icon.

The list view in EAC is designed to remove limitations that existed in ECP and EMC in Exchange 2010. The ECP is capable of listing up to only 500 objects in one page at the same time, and if you want to view objects that aren't listed in the Details pane, you need to use Search and Filter options to find those specific objects. In Exchange 2013, the viewable limit from within the EAC list view is approximately 20,000 objects for on-premises deployments and 10,000 objects in Exchange Online. In addition, paging is included so you can page to the results. In the Recipients list view, you can also configure page size and export the data to a CSV file.

When you select an object from the list view, information about that object is displayed in the Details pane. In some cases (for example, with recipient objects), the Details pane includes quick management tasks. For example, if you navigate to Recipients and then Mailboxes, and select a mailbox from the list view, the Details pane displays an option to enable or disable the archive for that mailbox. The Details pane can also be used to bulk-edit several objects. Simply press the CTRL key, select the objects you want to bulk-edit, and use the options in the Details pane. For example, selecting multiple mailboxes allows you to bulk-update users' contact and organization information, custom attributes, mailbox quotas, Outlook Web App settings, and more.

■ **Note** Supported browsers for the EAC are Internet Explorer 8 or later, Firefox 11 or later, Safari 5.1 or later, and Chrome 18 or later.

Exchange Management Shell

The Exchange Management Shell (EMS) is the core of Exchange Server management and this is what this book is all about. This is the place where you can configure everything—every little, tiny tidbit of Exchange Server. The EMS is not new; its first version appeared with Exchange Server 2007 and EMS has become more and more important over the years.

EMS is running on top of the Windows PowerShell and as such it can use all functionality that's available in PowerShell, like pipelining, formatting output, saving to local disk, ordering the output, or using filtering techniques. We'll discuss the most important basics here but also at various points throughout this book.

PowerShell

Lots of Microsoft server applications have their own management shell and all are running on top of Windows PowerShell; and whether you like it or not, PowerShell is the future for Windows management and for applications that run on top of Windows. And it's not only Microsoft that's using PowerShell for managing their applications; third-party vendors are also writing PowerShell add-ons for their products. Examples of these are HP, for their EVA storage management solutions; VMware, for their virtualization platform; and KEMP, for their load-balancing solutions.

The first version of PowerShell was a downloadable add-on for Windows 2003, but Windows Server 2008 was the first operating system that came with PowerShell built into the product.

PowerShell is a command-line shell and scripting environment, and it uses the power of the .NET Framework. But PowerShell is not text based, it is object based and as such it supports nice features such as pipelining, formatting, or redirecting the output. All objects have properties or methods and that's not different in PowerShell.

The last feature we're going to discuss is additional modules, such as the Server Manager, Active Directory, and the Exchange module.

Object Model

Although a command-line interface, PowerShell uses an object-oriented model. This means you are working with objects and not with normal text, as in a regular command prompt.

Since an object is returned, it can be manipulated or you can check certain attributes. For example, you can request information regarding an Exchange server with the following command:

```
Get-ExchangeServer -Identity AMS-EXCH01
```

Although it is returned as text on the console, it is actually an object being returned and you can treat it this way; for example:

```
(Get-ExchangeServer -Identity AMS-EXCH01).AdminDisplayVersion
```

This will return the AdminDisplayVersion property of the Exchange server. Or, when moving mailboxes and you want to check the number of mailboxes that are in the queue waiting to be processed, you can use the following command:

```
(Get-MoveRequest -MoveStatus Queued).count
```

So, the output of a command is an object and you can continue working with this object. This way you can use the output of one command as actual input for another command, a technique which is called pipelining. This technique is very often used in managing Exchange environments.

Pipeline

You can see a pipeline as a series of connected segments of pipe where all items or objects pass through each segment. Each segment has its own functionality or purpose and can alter the objects. To create a new pipeline the pipe operator “|” is used with the various commands. The simplest form of a pipeline is to use a Get command in conjunction with a Set command; for example:

```
Get-Mailbox | Set-Mailbox
```

In this command a pipeline is created between the Get-Mailbox and the Set-Mailbox commands. The Get-Mailbox command retrieves one or more mailbox objects, and these objects are sent through the pipeline to the Set-Mailbox command, which can make certain changes to the mailbox objects.

Personally, I use this pipelining a lot when administering Exchange server. You can use the various Get commands to retrieve objects from Exchange and you can actually see if you have the right objects. Then you can pipe them into the corresponding Set command and you’re done. Very valuable!

Objects and Members

Each object in PowerShell has members, and members can be properties or methods. A property is something that has a value—for example, the name of a mailbox. A method is something that can be executed against an object—for example, to clone a mailbox.

To see all members of a particular mailbox you can use the following command:

```
Get-Mailbox -Identity Administrator | Get-Member
```

This command will result in an output similar to:

```
[PS] C:\>Get-Mailbox -Identity Administrator | Get-Member
```

```
TypeName: Microsoft.Exchange.Data.Directory.Management.Mailbox
```

Name	MemberType	Definition
----	-----	-----
Clone	Method	System.Object Clone(), System.Object...
Equals	Method	bool Equals(System.Object obj)
GetHashCode	Method	int GetHashCode()
GetProperties	Method	System.Object[] GetProperties(System.Collections...
GetProxyInfo	Method	System.Object GetProxyInfo(), System.Object...
GetType	Method	type GetType()
ResetChangeTracking	Method	void IConfigurable.ResetChangeTracking()
SetProxyInfo	Method	void SetProxyInfo(System.Object proxyInfoValue)...
PSComputerName	NoteProperty	System.String PSComputerName=ams-exch01.contoso.com
PSShowComputerName	NoteProperty	System.Boolean PSShowComputerName=False
RunspaceId	NoteProperty	System.Guid RunspaceId=2e837bce-b1a8-4004-...
AcceptMessagesOnlyFrom	Property	Microsoft.Exchange.Data.MultiValuedProperty...
AddressBookPolicy	Property	Microsoft.Exchange.Data.Directory.ADObjectId...

AddressListMembership	Property	Microsoft.Exchange.Data.MultiValuedProperty...
AdminDisplayVersion	Property	Microsoft.Exchange.Data.ServerVersion...
ArchiveDatabase	Property	Microsoft.Exchange.Data.Directory.ADObjectId
ArchiveGuid	Property	guid ArchiveGuid {get;}
ArchiveName	Property	Microsoft.Exchange.Data.MultiValuedProperty[string]...

When you use PowerShell to retrieve an object, only a limited set of members is shown. This is purely practical; your console would be overwhelmed with data if all members were shown.

Formatting

It is possible to format the output as shown on the console using cmdlets that start with the *Format* verb. The following are used throughout this book:

- **Format-List.** This is abbreviated to FL and is used to show all properties of a certain object. To retrieve all properties of the Administrator mailbox you would use `Get-Mailbox -Identity | FL`
- **Format-Table.** This is abbreviated to FT and can be used to retrieve certain properties of an object. The `Get-Mailbox` command, for example, returns only the Name, Alias, ServerName, and ProhibitSendQuota properties. To retrieve the Name, Alias, Database, and ArchiveState, a command similar to `Get-Mailbox -Identity Administrator | FT -Property Name, Alias, Database, ArchiveState` can be used.
- **Format-Wide.** This command is abbreviated to FW and shows only one property of an object. Typically it shows only the default property of an object; for a mailbox this would be its name, but you can select another property using the `-Property` option.

In addition to these *Format* verbs you can use the `-Wrap` and `-AutoSize` parameters in PowerShell to format the output of the `Format-Table` command, as shown on the console. The `-Wrap` option does not truncate output in a column, but it wraps all output in its column, thereby showing the entire property. The `-AutoSize` option varies the width of the column, depending on the data that is shown in the column.

Important to note is that PowerShell expects the first column to be the most important, decreasing the importance with subsequent columns. As such, later columns can even be removed from the output when there's too much information to be shown. If this happens, you can change the order of information shown on the console by reordering the properties using the `-Property` option.

Normally the output of commands is shown on the console. It is possible to redirect the output elsewhere using the *Out* verb in PowerShell. The following options are available in PowerShell.

- **Out-Host.** The `Out-Host` option redirects the output to the console, which is the default option. You can use the `-Paging` option to show only a limited amount of information at one time. You can use the `<SPACE>` to view another page with information on the console.
- **Out-Null.** The `Out-Null` option immediately discards any information without showing it on the console. However, any error message or, more specifically, output from the error stream will be shown on the console.
- **Out-Printer.** The `Out-Printer` option redirects any output directly to the printer. The default printer is used if no printername is provided; otherwise, a command similar to `Out-Printer -Name "HP LaserJet 1200 Series PCL 5"` can be used.
- **Out-File.** The `Out-File` command is often used because it redirects any output to a (Unicode) file on the local hard disk. If a pure ASCII-coded file is needed, the `-Encode ASCII` option can be used—for example, `Out-File -FilePath C:\Logging\Mailboxes.txt -Encode ASCII`.

These methods and concepts are widely referenced throughout this book.

Grouping

Another useful parameter to organize output is the `GroupBy` control. Long output listings that are hard to view offer the option to group the output based on a property. For example, it is possible to retrieve all users from Active Directory and group the output by the value of their `Company` attribute, like this:

```
Get-User | Format-Table -Property Name,SamAccountName,Company -Sort Company -GroupBy Company
```

Filtering

It's also possible to filter the output of the `Get-User` command with the `-Filter` parameter. For example, to mailbox-enable all users whose company attribute is set to "Fourth Coffee," enter the following command:

```
Get-User -Filter {Company -eq "Fourth Coffee"}
```

■ **Note** Whenever possible you should use the `-Filter` option. This will only send the objects to the console that come out of the filter, resulting in much more efficient processing.

If you want to be even more specific—for example, to mailbox-enable all users whose company attribute is set to "Fourth Coffee" *and* whose department attribute is set to "Marketing," enter the following command:

```
Get-User -Filter {(Company -eq "Fourth Coffee") -AND (Department -eq "Marketing")}
```

In short, the following operations are available for the `-Filter` option:

- `-and`
- `-or`
- `-not`
- `-eq` (equals)
- `-ne` (does not equal)
- `-lt` (less than)
- `-gt` (greater than)
- `-like` (compare strings by using wildcard rules)
- `-notlike` (compare strings by using wildcard rules)

Conversion

It's possible to convert objects to a certain format—for example, HTML or CSV. This can be useful if you want to just collect data or you want to process it using applications like Excel. For instance, to collect a simple list of mailboxes and export that information to a CSV file that you can import in Excel, enter the following cmdlet:

```
Get-Mailbox | Select DisplayName, WindowsEmailAddress | Export-CSV -NoTypeInfo Mailboxes.csv
```

When exporting output into a CSV file, PowerShell writes the Type of Object onto the first line of the CSV file—something like `#TYPE Selected.Microsoft.Exchange.Data.Directory.Management.Mailbox`, as shown in Figure 1-8.

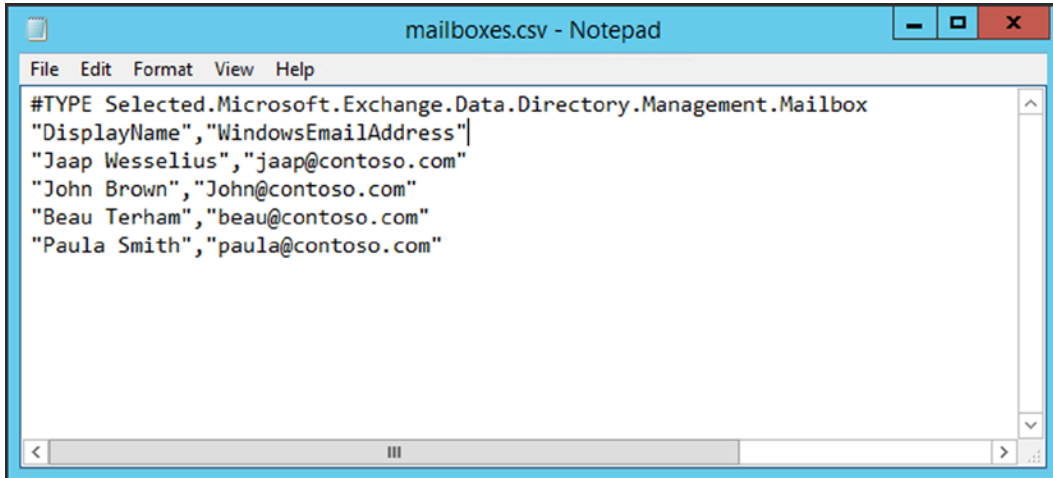


Figure 1-8. The `Export-CSV` creates an additional `#Type` entry in the CSV file

The `-NoTypeInfo` option prevents this line in the output file.

Conversion cmdlets also have an import counterpart—that is, `Import-CSV`—which allows you to import information that is stored in a certain format.

Variables

Using variables is not really specific for PowerShell, but every command-line interface or scripting engine can use variables, and that's no different in PowerShell.

As explained earlier, PowerShell is using objects, and you can store objects in variables so you can use them later on. This can be used in a PowerShell script, but also on the command line. An object is kept alive as long as the PowerShell window is open or until it is destroyed.

Variables are identified with a `$` character, followed by any name you want. Of course, it is good practice to use an easy-to-identify name.

To create a variable called `$AdminMailbox` and store the Administrator mailbox object in it, you can use the following command:

```
$AdminMailbox = Get-Mailbox -Identity Administrator
```

The mailbox object is now stored in the variable `$AdminMailbox` and ready for use directly, or for later use. To view the contents of the variable, you can type in its name in the PowerShell window.

```
[PS] C:\Windows\system32>$AdminMailbox
```

Name	Alias	ServerName	ProhibitSendQuota
-----	-----	-----	-----
Administrator	Administrator	ams-exch01	Unlimited

```
[PS] C:\Windows\system32>
```