Chapter 8: Managing Modern Public Folders

Gareth Gudger

In this chapter, we look at modern public folders. You will learn about the following aspects of public folders:

- Public folder mailboxes
- Public folder hierarchy
- Public folders
- Mail-enabling a public folder

In addition, we discuss how to perform common administrative tasks through both the Exchange Admin Center (EAC) and the Exchange Management Shell (EMS).

A brief history on public folders

Public folders were first introduced in 1996 with the release of Exchange 4.0. The premise behind public folders was to create a collaborative space that could be easily accessed through a common messaging client. Public Folders were frequently used for sharing calendars, contacts, forms, lists, documents and other types of data. For example, it was not uncommon to see a folder tree full of business contacts, or, a company calendar for tracking corporate events and holidays.

This data was housed inside a series of public folders which in turn was stored in a public folder database. Like the mailbox database, the public folder database leveraged the Extensible Storage Engine (ESE) and was managed by the Information Store. Where they differed was that public folders maintained their own multi-master replication model. This allowed content in one public folder database to be replicated to a public folder database on another server. The replication properties were also granular, allowing the admin to define which public folders should be replicated, which servers should host a copy, and the replication schedule for each folder. This was especially useful for organizations that had multiple locations over slow and costly WAN links as it allowed relevant public folder data for that location to be stored closer to the users it served. The hierarchy of the public folder could also be replicated along with content to each public folder database.

With the release of Exchange 5.5 public folders gained the ability to be mail-enabled. This allowed a public folder to send and receive mail. It was not uncommon to see a public folder used in a similar fashion to that of a shared mailbox. For example, a mail-enabled public folder may have been used as the target for all inquiries from the company website.

In early versions of Exchange public folders also contained many hidden system folders. These folders, referred to as the NON-IPM Subtree, were used for various tasks including the delivery of the Offline Address Book (OAB) and a user's free/busy information. This dependency changed in Exchange 2010 where many of the system folder functions were rolled into Exchange Web Services (EWS).

Exchange 2013 introduced the modern public folder. Modern public folders were a complete rewrite of the underlying public folder architecture. Public folder content and the hierarchy were moved from a dedicated public folder database and into public folder mailboxes. These mailboxes were in turn stored in regular databases, allowing them to coexist with other mailboxes types, such as user mailboxes. Public folder replication was also eliminated, allowing public folder content to be made highly available through the Database Availability Group (DAG).

With the release of Exchange 2013 all public folders from prior versions were now rebranded as legacy public folders. To the end user the experience was the same. To the admin modern public folders leveraged the same storage and replication technologies that any other mailbox type would use. This simplified life for the Exchange admin as now only one storage and one replication type was managed instead of two.

With the initial release of Exchange 2016 the maximum number of public folders mailboxes was 100. With Exchange 2016 CU2 the maximum number of public folder mailboxes increased to 1,000. With future CU's it is likely this and other limits will continue to climb.

With the possibility of 1,000 public folder mailboxes, and a maximum size of 100 GB each, this allows for a theoretical maximum of 100,000 GB of public folder data. It is worth noting that Microsoft recommends to split public folder mailboxes once they reach 50% capacity.

Other limits include no more than 1 million folders in the hierarchy, a limit of 1 million items per folder, and a folder depth not to exceed 300 levels.

It is strongly advised not to exceed these limits. Exceeding these limits puts your environment in an unsupported configuration. For more information on public folder limits be sure to check this article: https://technet.microsoft.com/en-us/library/dn594582(v=exchg.160).aspx

Note: For information on how to migrate public folders from a previous version of Exchange check *Chapter 13: Updating and Migration*.

Public folder mailboxes

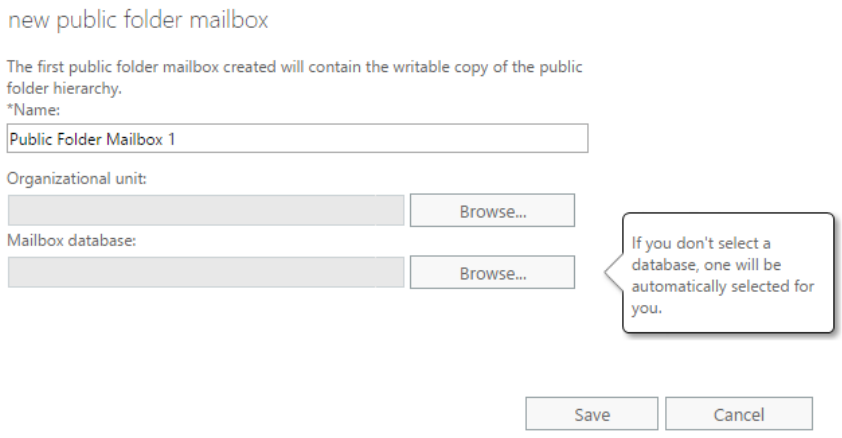
As mentioned public folders are stored inside a special type of mailbox referred to as a public folder mailbox. Before we create a public folder at least one public folder mailbox must exist. The first public folder mailbox in the environment also hosts the primary hierarchy. We will cover more on the hierarchy in a later section.

Creating a public folder mailbox

In this section, we will look at how to create a public folder mailbox using both the Exchange Admin Center (EAC) and the Exchange Management Shell (EMS). Let's explore using the EAC first.

From the EAC select **Public Folders** on the left and **Public Folder Mailboxes** across the top. From here click the **New Public Folder Mailbox** button (represented by a plus sign).

On the *New Public Folder Mailbox* window (Figure 8-1) type a **Name** for the mailbox. In our example, we have typed *Public Folder Mailbox 1*. Optionally you can specify an organizational unit and database. Click **Save**.



new public folder mailbox

The first public folder mailbox created will contain the writable copy of the public folder hierarchy.

*Name: Public Folder Mailbox 1

Organizational unit:

Mailbox database:

If you don't select a database, one will be automatically selected for you.

Figure 1: Creating a public folder mailbox

To perform the same task in EMS, we will use the **New-Mailbox** cmdlet with the **-PublicFolder** switch. In this example, we will create a second public folder mailbox named *Public Folder Mailbox 2*.

```
[PS] C:\> New-Mailbox -PublicFolder -Name "Public Folder Mailbox 2"

Name                       Alias                 ServerName          ProhibitSendQuota
----                       -----                 ----------          -----------------
Public Folder Mailbox 2    PublicFolderMailbox2  exch01              Unlimited
```

In this command, **-PublicFolder** identifies that the type of mailbox we are creating is a public folder mailbox. Without this parameter, the **New-Mailbox** cmdlet would create a user mailbox. As it implies the **-Name** parameter gives our public folder mailbox a name. When you create a public folder mailbox it creates a disabled account in Active Directory. Should we later mail-enable our public folder all mail attributes will be stored in this disabled account. You can specify the location of this account using the **-OrganizationalUnit** parameter. The **-Database** parameter allows us to specify which database should host the mailbox.

To locate all public folder mailboxes, we can use **Get-Mailbox** cmdlet with the **-Public** parameter.

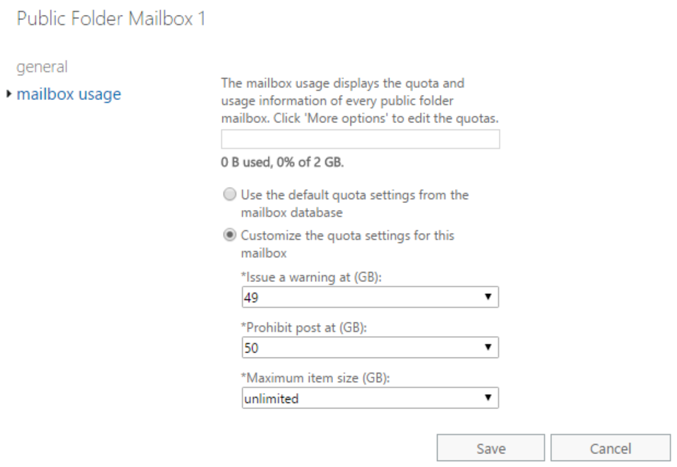
```
[PS] C:\> Get-Mailbox -Public | Format-Table

Name                       Alias                 ServerName    ProhibitSendQuota
----                       -----                 ----------    -----------------
Public Folder Mailbox 1    PublicFolderMailbox1  exch01        Unlimited
Public Folder Mailbox 2    PublicFolderMailbox2  exch02        Unlimited
```

Managing a public folder mailbox

The public folder mailbox possesses very few properties to manage. From the EAC select a public folder mailbox and click **Edit** button (represented by a pencil). On the **General** tab, we only have the option to rename the mailbox. If we switch over to the **Mailbox Usage** tab, we can view how much space we are currently consuming. If we click the **More options** link we can modify the quota for the public folder mailbox. By default, this is set to **Use the default quota settings from the mailbox database**. However, much like a regular user mailbox we can select **Customize the quota settings for this mailbox** and specify a different value.

In our example (Figure 7-2) we have specified to Issue a warning at 49 GB and prohibit the posting of new public folder items at 50 GB. Setting a quota helps us keep our public folder mailboxes under the 100 GB limit. Setting a value of 50 GB will allow us to keep those mailboxes at or below 50% utilization. If you do not set a quota on the public folder mailbox it will inherit the limits set by the database. By default, a database is set to prohibit a public folder mailbox (and all other mailbox types) from exceeding 2 GB.



Public Folder Mailbox 1

general

▸ mailbox usage

The mailbox usage displays the quota and usage information of every public folder mailbox. Click 'More options' to edit the quotas.

0 B used, 0% of 2 GB.

○ Use the default quota settings from the mailbox database

● Customize the quota settings for this mailbox

*Issue a warning at (GB): 49

*Prohibit post at (GB): 50

*Maximum item size (GB): unlimited

Figure 2: Public folder mailbox quota

To set a quota through EMS issue the following command.

```
[PS] C:\> Set-Mailbox -PublicFolder -Identity "Public Folder Mailbox 2" -ProhibitSendReceiveQuota
50GB -IssueWarningQuota 49GB -UseDatabaseQuotaDefaults $false
```

In this command:

-PublicFolder is required whenever you are modifying a public folder mailbox.

-**Identity** specifies the public folder mailbox to modify.

-ProhibitSendReceiveQuota translates to the **Prohibit Post** field in the EAC. Once this content of this mailbox exceeds the specified value no more content can be added to any public folders hosted by this mailbox. In addition, any mail-enabled public folders will not be able to send or receive mail.

-IssueWarningQuota specifies a value at which a warning will be issued to the mailbox. This value is required to be less than the prohibit send/receive value.

-UseDatabaseQuotaDefaults determines whether to follow the quota set by the database where the public folder belongs. If set to true, any custom values set in this command are ignored. If set to false, the custom values are in effect.

Move a public folder mailbox to a different database

Like a user mailbox we can move a public folder mailbox to a different database. This can be critical in situations where we need to move public folder content closer to the users it is serving. Another scenario is to help spread the load generated by public folders across as many databases as possible.

One item to note is that the public folder mailbox will become unavailable during the final stages of the move, so it may be best to perform this task during a maintenance window or postpone the completion stage for a later date. How long the move takes depends on the size of the mailbox and the current load of the server.

This is not a task we can perform in the EAC. Instead we need to use PowerShell. For example, to move *Public Folder Mailbox 2* to another database our command would look like this.

```
[PS] C:\> New-MoveRequest -Identity "Public Folder Mailbox 2" -TargetDatabase "DB02"

DisplayName                 StatusDetail          TotalMailboxSize            PercentComplete
-----------                 ------------          ----------------            ---------------
Public Folder Mailbox 2     WaitingForJobPickup   10.65 KB (10,902 bytes)     0
```

We can monitor the progress of the move with the **Get-MoveRequest** cmdlet.

```
[PS] C:\> Get-MoveRequest -Identity "Public Folder Mailbox 2"

DisplayName                                 Status                TargetDatabase
-----------                                 ------                --------------
Public Folder Mailbox 2                     Completed             DB02
```

For a verbose output, you can pipe **Get-MoveRequest** into **Get-MoveRequestStatistics**. The **-IncludeReport** pulls the log detail for the move.

```
[PS] C:\> Get-MoveRequest -Identity "Public Folder Mailbox 2" | Get-MoveRequestStatistics -
IncludeReport | Format-List
```

Once the move is completed, and you no longer need access to the move logs, you can remove the move request. For this you can use the **Remove-MoveRequest** cmdlet.

```
[PS] C:\> Remove-MoveRequest -Identity "Public Folder Mailbox 2" -Confirm:$false
```

Public folder hierarchy

The first public folder mailbox in the environment hosts the primary hierarchy. The primary hierarchy is the only writable copy of the hierarchy. The hierarchy contains information such as the folder structure, folder permissions and the location of folder content. The hierarchy does not contain information about a public folders mail attributes. Mail attributes for a mail-enabled public folder are stored in Active Directory in the disabled mailbox object.

Any additional public folder mailboxes host a secondary read-only copy of the hierarchy. This is useful as a geographically dispersed organization can place a second public folder mailbox with a read-only copy of the hierarchy closer to the user's location. This is also necessary where a single public folder mailbox cannot meet the I/O required of a large organization.

Changes to the hierarchy, such as a change to folder permissions, can only written against the primary hierarchy stored in the initial public folder mailbox. Should the database hosting the primary hierarchy mailbox become unavailable no changes can be made to the hierarchy (or any content stored in that mailbox). In addition, the primary hierarchy cannot be moved to another mailbox. Nor can a mailbox hosting a secondary hierarchy be promoted to a primary hierarchy. It is absolutely important to ensure the primary hierarchy mailbox be made highly available with a database availability group (DAG).

To determine which mailbox hosts our primary hierarchy we have a couple of commands at our disposal. First, we can use the **Get-Mailbox** cmdlet to retrieve the value of **IsRootPublicFolderMailbox**. In the example below *Public Folder Mailbox 1* has this value set to true identifying it as owning the primary hierarchy.

```
[PS] C:\> Get-Mailbox -Public | Format-Table Name, ExchangeGUID, IsRootPublicFolderMailbox

Name                          ExchangeGuid                                 IsRootPublicFolderMailbox
----                          ------------                                 -------------------------
Public Folder Mailbox 1       44d0ea27-e37f-4a05-8fdc-6001fcfd7a13         True
Public Folder Mailbox 2       f7728513-b3e9-4211-b349-b4b8a2c42f6c         False
```

Our second option is to run the **Get-OrganizationConfig** cmdlet. From here we look for the value of the **RootPublicFolderMailbox** field.

```
[PS] C:\> Get-OrganizationConfig | Format-List RootPublicFolderMailbox

RootPublicFolderMailbox: 44d0ea27-e37f-4a05-8fdc-6001fcfd7a13
```

To determine which mailbox this Exchange GUID references we can run the follow command.

```
[PS] C:\> Get-Mailbox 44d0ea27-e37f-4a05-8fdc-6001fcfd7a13 -Public

Name                          Alias                 ServerName     ProhibitSendQuota
----                          -----                 ----------     -----------------
Public Folder Mailbox 1       PublicFolderMailbox1  exch01         Unlimited
```

The hierarchy is replicated through a process called Incremental Change Synchronization (ICS). The process only synchronizes the items that have changed in the hierarchy rather than doing a full sync. Replication of the public folder hierarchy can occur at different intervals. The interval is based solely on public folder usage in the organization.

In summary:

- The creation of a new folder in the hierarchy triggers an immediate sync
- If users are actively using public folders changes are synchronized every 15 minutes
- If users are not using public folders changes are synchronized every 24 hours

If you wish to force an immediate public folder synchronization you can use the **Update-PublicFolderMailbox** cmdlet.

To troubleshoot sync operations, you can leverage the **Get-PublicFolderMailboxDiagnostics** cmdlet. The **-IncludeHierarchyInfo** parameter adds some extra metrics around the objects in the hierarchy. For example, to see how *Public Folder Mailbox 2* is performing we can issue the following command.

```
[PS] C:\> Get-PublicFolderMailboxDiagnostics -IncludeHierarchyInfo -Identity "Public Folder Mailbox
2"

SyncInfo      : IsFullSyncCompleted: False
                NextPushSyncMessage: 0
                NumberOfFoldersSynced: 0
                NumberOfFoldersToBeSynced: 0
                NumberOfBatchesExecuted: 0
                BatchSize: 0
                NumberofAttemptsAfterLastSuccess: 0

AssistantInfo : LastAttemptedSyncTime: 5/3/2017 1:39:00 PM
                LastSuccessfulSyncTime: 5/3/2017 1:39:10 PM
                FirstFailedSyncTimeAfterLastSuccess: 12/31/1600 7:00:00 PM
                NumberofAttemptsAfterLastSuccess: 0
                LastSyncCycleLog: (report omitted for space)

HierarchyInfo : TotalFolderCount: 1
                MailPublicFolderCount: 0
                MaxFolderChildCount: 0
                HierarchyDepth: 0
                CalendarFolderCount: 0
                ContactFolderCount: 0
                InfoPathFolderCount: 0
                JournalFolderCount: 0
                NoteFolderCount: 0
                StickyNoteFolderCount: 0
                TaskFolderCount: 0
                OtherFolderCount: 1
```

Unlike the hierarchy, the content inside public folders is not replicated. The content is only stored in a single mailbox. It is never replicated between mailboxes. The hierarchy will direct all users to the same public folder mailbox that hosts the content, irrespective of where the user is located. That said if certain content is intended for users in a specific geography it is wise to get that content into a public folder mailbox in close proximity to those users. We will cover moving public folders between mailboxes in a later section.

By default, any public folder mailbox can serve a copy of the hierarchy to any user. Exchange load balances user connections across all public folder mailboxes in the environment. There may be a need to prevent this behaviour. For example, if a public folder mailbox is situated across a WAN link we may want to govern which users that public folder mailbox will serve.

In the following example, *Public Folder Mailbox 2* is located across a WAN link in the remote New York site. The New York site comprises 30% of the user base. The remaining 70% of users are located in Florida. We want to prevent *Public Folder Mailbox 2* from serving the hierarchy to our Floridian users.

To accomplish this task, we must first exclude this mailbox from serving the hierarchy. To do this we set the **-IsExcludedFromServingHierarchy** parameter to *$true*.

```
[PS] C:\> Set-Mailbox -PublicFolder -Identity "Public Folder Mailbox 2" -
IsExcludedFromServingHierarchy $true
```

This blocks the public folder mailbox from serving the hierarchy to any user that does not have a default public folder mailbox set. By default, no users will have a default public folder mailbox set. Users that already have *Public Folder Mailbox 2* set as their default public folder mailbox will still be served a hierarchy by this mailbox.

To assign this public folder mailbox to our remote users we use the **Set-Mailbox** cmdlet in conjunction with the **-DefaultPublicFolderMailbox** parameter. For example, to set user Joe Diaz to use the hierarchy from *Public Folder Mailbox 2* our command would resemble this.

```
[PS] C:\> Set-Mailbox -Identity "Joe Diaz" -DefaultPublicFolderMailbox "Public Folder Mailbox 2"
```

For multiple users, we can pipe **Get-Mailbox** into **Set-Mailbox**.

```
[PS] C:\> Get-Mailbox | Where {$_.OrganizationalUnit -eq "space-corp.net/Employees/New York/"} |
Set-Mailbox -DefaultPublicFolderMailbox "Public Folder Mailbox 2"
```

We could take this one step further by assigning our Floridian users a default public folder mailbox as well. For example, *Public Folder Mailbox 1*.

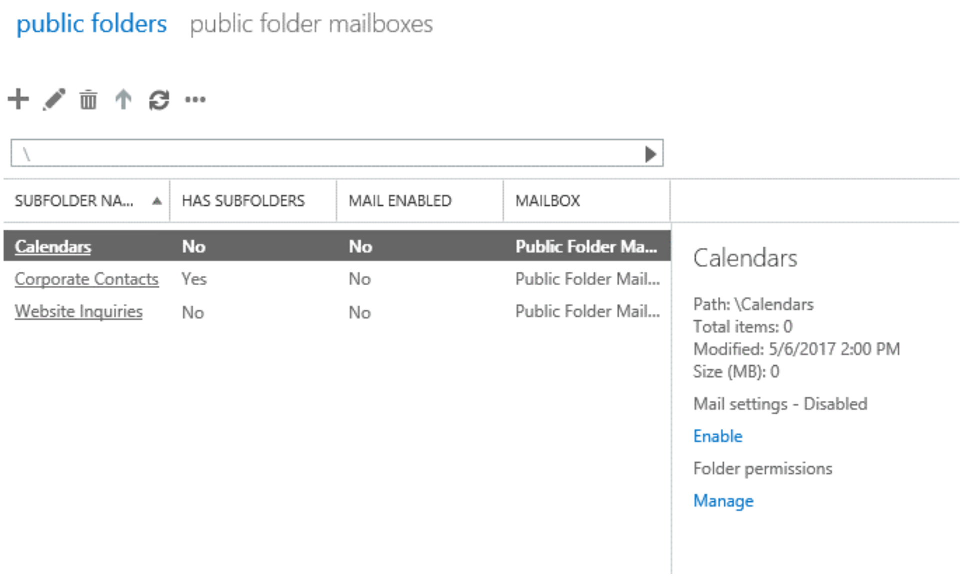
> **Real World:** Excluding a public folder mailbox from serving the hierarchy may also be useful for mailboxes that host popular content. That way a heavily accessed public folder mailbox is only tasked with serving content and not hierarchy. In contrast, if you manage a large public folder infrastructure you should consider moving all content out of the primary hierarchy mailbox. That way the primary mailbox is only tasked with hierarchy operations.

Public folders

Now that we have our public folder mailboxes in place let's switch gears and look at public folders. Public folders are the containers our users see. Public folders can be used for sharing calendars, contacts, email messages and other document types. Public folders can be managed from the EAC by navigating to the **Public Folders** tab on the left followed by the **Public Folders** tab across the top.

From this screen (Figure 8-3) we can see all top-level public folders in our environment. The columns on this screen provide some basic information such as whether this folder contains subfolders, if the folder has been mail enabled and what public folder mailbox currently hosts this folder.

From Figure 8-3 we can see we have three top level folders. The *Corporate Contacts* folder is identified as containing subfolders. However, none of the folders are mail-enabled and all folders are hosted by *Public Folder Mailbox 1*.



\

SUBFOLDER NA... ▲	HAS SUBFOLDERS	MAIL ENABLED	MAILBOX
Calendars	No	No	Public Folder Ma...
Corporate Contacts	Yes	No	Public Folder Mail...
Website Inquiries	No	No	Public Folder Mail...

Calendars

Path: \Calendars
Total items: 0
Modified: 5/6/2017 2:00 PM
Size (MB): 0

Mail settings - Disabled

Enable

Folder permissions

Manage

Figure 3: Managing Public Folders

To navigate through the folder tree, you can either type the full path to the folder in the search box, or, click the underlined folder. Selecting anywhere else on the row will only select the parent folder for editing. In many ways, the search bar operates like the breadcrumbs you may see on a website. The search bar will update the path as you navigate between different folders. When the search box only displays a single backslash, you are back at the top most folder level or root.

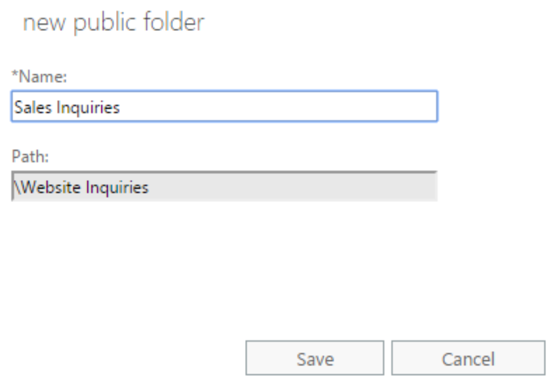
To navigate up one level click the **Go to parent folder button** (represented by an up-arrow). The up-arrow greys out once you reach the root. To navigate to the root with a single click, select the **Clear Search** button in the search bar (represented by a triangle).

Creating a public folder

In this section, we will look at how to create a public folder using both the EAC and EMS. Let's explore using the EAC first.

From the EAC select **Public Folders** on the left and **Public Folders** across the top. This will display all public folders currently in our environment. If you want to create a public folder in the root click the **New Public Folder** button (represented by a plus sign). If you want to create a public folder as a subfolder you must first navigate to the intended parent folder and then click the **New Public Folder** button. In our example, we are going to create a new public folder under our existing *Website Inquiries* folder (Figure 8-3). We will navigate to that first and then click new public folder.

On the *New Public Folder* window (Figure 8-4) type a **Name** for the public folder. In our example, we have typed *Sales Inquiries*. The **Path** field is read-only. This identifies where you were in the public folder tree when you clicked the new public folder button. Confirm this path is correct. If not click cancel. Otherwise click **Save**. After the dialog closes, be sure to click the **Refresh** button to display the new folder.



new public folder

*Name: Sales Inquiries

Path: \Website Inquiries

Figure 4: Creating a new public folder

To perform the same task in EMS, we will use the **New-PublicFolder** cmdlet. In this example, we will create a public folder named *Press Release Packets* under our existing *Website Inquiries* folder.

```
[PS] C:\> New-PublicFolder -Name "Press Release Packets" -Path "\Website Inquiries"

Name                                 Parent Path
----                                 -----------
Press Release Packets                \Website Inquiries
```

We can confirm the creation of this public folder with the **Get-PublicFolder** cmdlet.

```
[PS] C:\> Get-PublicFolder -Identity "\Website Inquiries" -GetChildren

Name                                 Parent Path
----                                 -----------
Press Release Packets                \Website Inquiries
Sales Inquiries                      \Website Inquiries
```

If you would rather see the public folder tree in its entirety you can run the following.

```
[PS] C:\> Get-PublicFolder -Recurse -ResultSize Unlimited
```

Unlike **-GetChilditem** which returns only the subfolders at the current folder level **-Recurse** returns all levels.

Mail-enabling a public folder

Mail-enabling a public folder allows that public folder to receive email. Depending on the permissions set the source of those emails could be both internal and external. Similar to shared mailboxes a mail-enabled public folder can act as a repository for users who need to collaborate on emails. A mail-enabled public folder could be the target for all inquiries from the company website. For example, info@space-corp.net could be assigned to a public folder. Any emails sent to that address would be delivered to this public folder.

To mail-enable a public folder, select the row for that public folder and click the **Enable** link in the task pane. You will receive a warning asking you to confirm the mail-enablement on the public folder. Click **Yes**. After a few seconds the folder will become mail-enabled. This will be represented by *Mail Enabled* column switching to *Yes* and the task pane reporting *Mail Settings – Enabled***.** The enable link will now display a disable option instead.

By default, the email address assigned to the mailbox will use the public folders name. Any spaces in that name will be removed. For example, a public folder named *Website Inquiries* would take on the email address websiteinquiries@space-corp.net. You can change the email address assigned to the public folder by editing the properties of the public folder. We will cover that in the next section.

Mail-disabling the public folder is a repeat of this process but clicking **Disable** instead. You will be prompted to confirm. Clicking **Yes** will remove all mail attributes and email addresses from the public folder.

To perform this task in EMS we use the **Enable-MailPublicFolder** cmdlet. In this example, we will mail-enable a public folder named *Website Inquiries*.

```
[PS] C:\> Enable-MailPublicFolder -Identity "\Website Inquiries"
```

In this command, we specify the path to the public folder with the **-Identity** parameter. You will notice we precede the folder name with a backslash. The first backslash identifies the root of the public folder tree. *Website Inquiries* is a top-level folder in the public folder hierarchy. If we wanted to mail-enable a subfolder several levels deep our command might look like this instead.

```
[PS] C:\> Enable-MailPublicFolder -Identity "\Website Inquiries\Sales Inquiries\New York Office"
```

This command will not return a response to the console. We can verify our work by running **Get-MailPublicFolder**.

```
[PS] C:\> Get-MailPublicFolder

Alias                Identity
-----                --------
WebsiteInquiries     space-corp.net/Microsoft Exchange System Objects/Website Inquiries
```

To allow external sources to email this public folder we also need to configure anonymous permissions on this public folder. Specifically, we need to allow anonymous users the ability to create items in this folder. For this task, we will use the **Add-PublicFolderClientPermission** cmdlet.

```
[PS] C:\> Add-PublicFolderClientPermission -Identity "\Website Inquiries" -AccessRights CreateItems
-User Anonymous

FolderName           User                 AccessRights
----------           ----                 ------------
Website Inquiries    Anonymous            {CreateItems}
```

Should we need to mail-disable a public folder we can run the following command.

```
[PS] C:\> Disable-MailPublicFolder -Identity "\Website Inquiries" -Confirm:$false
```

Managing public folder properties

To manage a public folder, select the row for that public folder and click **Edit** button (represented by a pencil). If you clicked on the public folder's name it will actually navigate you down one level. You can also double click the row to edit the properties.

The number of tabs you see will depend on whether this public folder has been mail-enabled or not. For a mail-enabled public folder you will see eight tabs. A public folder that has not been mail-enabled will only see the three.

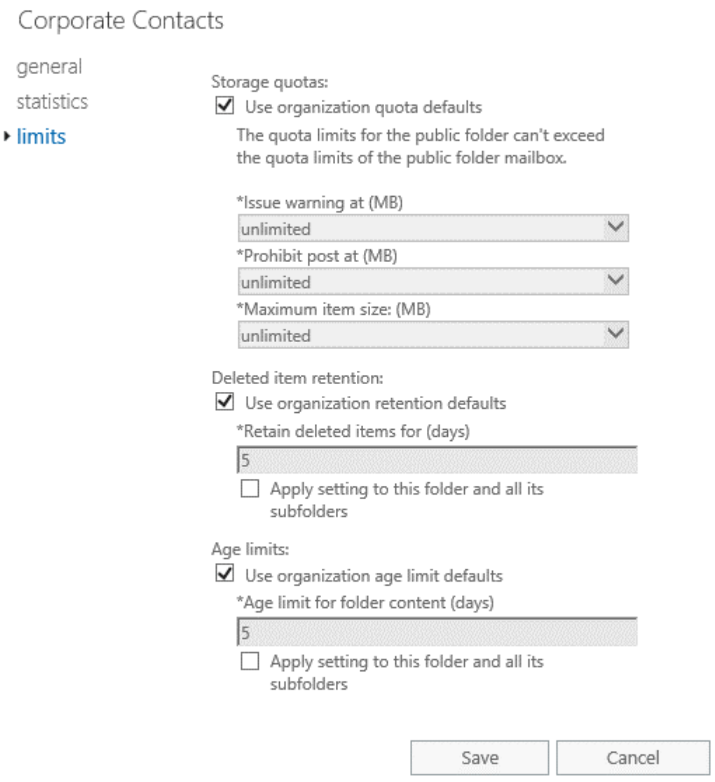
On the **General** tab, we only have the option to rename the public folder. All other fields are informational only. These fields include information such as the total items in the folder, the size of the folder in megabytes, which public folder mailbox hosts the public folder and the date the public folder was last modified.

The checkbox **Maintain per-user read and unread information** determines when an item is read if it should be marked read for all users, or, just the user that viewed the item. By default, this box is checked. If you uncheck this box, the first user to read the item will switch the status of the item to read for everyone else. The checkbox **Apply the read and unread setting to this folder and its subfolders** propagates the setting of the previous checkbox to all subfolders.

The **Statistics** tab is entirely read only. This tab lists the number of items in the public folders dumpster, the space consumed by that dumpster and various other metrics.

The **Limits** tab (Figure 8-5) allows you to define storage quotas and deleted item retention.

By default, the storage quota is set to **Use organization quota defaults** but you can uncheck this box and set specific values for just this public folder. These values govern how large the public folder can get and the maximum size for an individual item. This is separate from the quota set on a public folder mailbox. For example, you could dictate that a public folder mailbox should not exceed 50 GB and that this public folder should not exceed 5 GB.



Storage quotas:

☑ Use organization quota defaults

The quota limits for the public folder can't exceed the quota limits of the public folder mailbox.

*Issue warning at (MB): unlimited

*Prohibit post at (MB): unlimited

*Maximum item size: (MB): unlimited

Deleted item retention:

☑ Use organization retention defaults

*Retain deleted items for (days): 5

□ Apply setting to this folder and all its subfolders

Age limits:

☑ Use organization age limit defaults

*Age limit for folder content (days): 5

□ Apply setting to this folder and all its subfolders

Figure 5: Managing public folder limits

The deleted item retention settings determine how long a deleted item is retained. By default, this is inherited from the organization level, which is 5 days. To set a custom value deselect the **Use organization retention defaults** checkbox and specify a value in days in the **Retain deleted items** field. You can also choose to apply this setting to all subfolders.

The age limit settings determine how long an item is kept in the public folder. Unlike a user mailbox a public folder mailbox can't be assigned a retention policy. Instead public folders can either use an organizational setting to automatically delete items over a certain age, or, an aging value can be set on a folder by folder basis. To configure a custom age limit for a public folder, deselect the **Use organization age limit** checkbox and specify a value in days in the **Age limit for folder content** field. You can also choose to apply this setting to all subfolders.

To configure these options with EMS we use the **Set-PublicFolder** cmdlet. For example, to set a quota warning and maximum size on the *Corporate Contacts* public folder we can issue the following command.

```
[PS] C:\> Set-PublicFolder -Identity "\Corporate Contacts" -IssueWarningQuota 4GB -ProhibitPostQuota
5GB
```

To specify how large an individual item can be we can use the **-MaxItemSize** parameter. In the following example, we restrict individual items to 3 MBs.

```
[PS] C:\> Set-PublicFolder -Identity "\Corporate Contacts" -MaxItemSize 3MB
```

We can check the settings of a public folder by using **Get-PublicFolder**.

```
[PS] C:\> Get-PublicFolder "\Corporate Contacts" | Format-List

Identity                      : \Corporate Contacts
Name                          : Corporate Contacts
MailEnabled                   : False
MailRecipientGuid             : 00000000-0000-0000-0000-000000000000
ParentPath                    : \
LostAndFoundFolderOriginalPath:
ContentMailboxName            : Public Folder Mailbox 1
ContentMailboxGuid            : 44d0ea27-e37f-4a05-8fdc-6001fcfd7a13
EformsLocaleId                :
PerUserReadStateEnabled       : True
EntryId                       :
DumpsterEntryId               :
ParentFolder                  :
OrganizationId                :
AgeLimit                      :
RetainDeletedItemsFor         :
ProhibitPostQuota             : 5 GB (5,368,709,120 bytes)
IssueWarningQuota             : 4 GB (4,294,967,296 bytes)
MaxItemSize                   : 3 MB (3,145,728 bytes)
LastMovedTime                 :
AdminFolderFlags              :
FolderSize                    : 0
HasSubfolders                 : True
FolderClass                   : IPF.Note
FolderPath                    : {Corporate Contacts}
AssociatedDumpsterFolder      :
DefaultFolderType             : None
ExtendedFolderFlags           :
```

We can also use the **Get-PublicFolderStatistics** command to pull metrics for all folders in the environment.

```
[PS] C:\> Get-PublicFolderStatistics | Format-Table Name, ItemCount, TotalItemSize -AutoSize
```

> **Note:** For an extensive list of all available parameters for **Set-PublicFolder** cmdlet check the following article https://technet.microsoft.com/en-us/library/aa998596(v=exchg.160).aspx

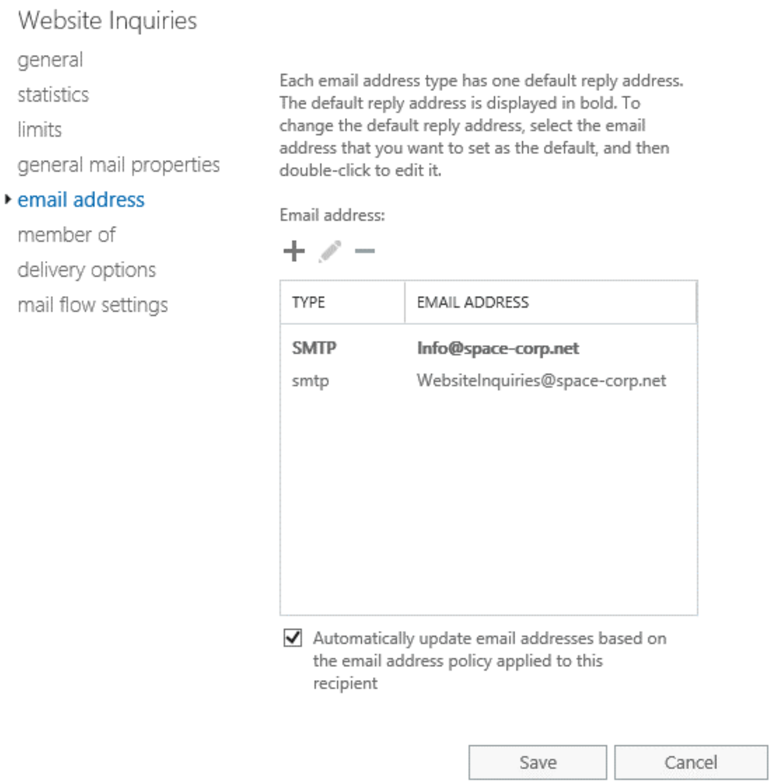
Managing public folder mail settings

If a public folder has been mail-enabled the properties of that folder will contain five additional tabs. We covered the first three tabs in the last section so let's jump straight to **General Mail Properties**.

The general mail properties tab allows us to modify the **Alias** and the **Display Name** of the public folder. In most cases changing the alias will alter the primary email address. The exception to this rule is if the public folder is configured not to use the email address policy. When the public folder is configured to use the email address policy it will always match against the default email address policy.

Selecting the **Edit** button allows you to add up to 15 custom attributes. These attributes can be useful when developing filters for custom address lists. For example, you could create a custom address list that matches *Custom Attribute 1* having the value of *Corporate Contacts*. All public folders with this attribute (and any other mail objects) will appear in that address list.

The **Email Address** tab (Figure 8-6) lists all addresses assigned to the public folder. By default, addresses populated here are assigned by the default email address policy. Deselecting the checkbox **Automatically update email addresses based on the email address policy** will block this public folder from retrieving its addresses from the default email address policy. Keep in mind that any addresses already assigned by a policy will not be removed when this checkbox is unchecked.



Each email address type has one default reply address. The default reply address is displayed in bold. To change the default reply address, select the email address that you want to set as the default, and then double-click to edit it.

Email address:

TYPE	EMAIL ADDRESS
SMTP	**Info@space-corp.net**
smtp	WebsiteInquiries@space-corp.net

☑ Automatically update email addresses based on the email address policy applied to this recipient

Figure 6: Managing email address for a public folder

The public folder's primary address is highlighted in bold. This is the address people see when emails are sent from the public folder. The email address policy governs the primary address. To change the primary address, you will first need to deselect **Automatically update email addresses based on the email address policy**. Then select the address you want to make the primary address and click the **Edit** button. Check the box **Make this the reply address**. Click **Ok**. This address now becomes the public folder's primary address.

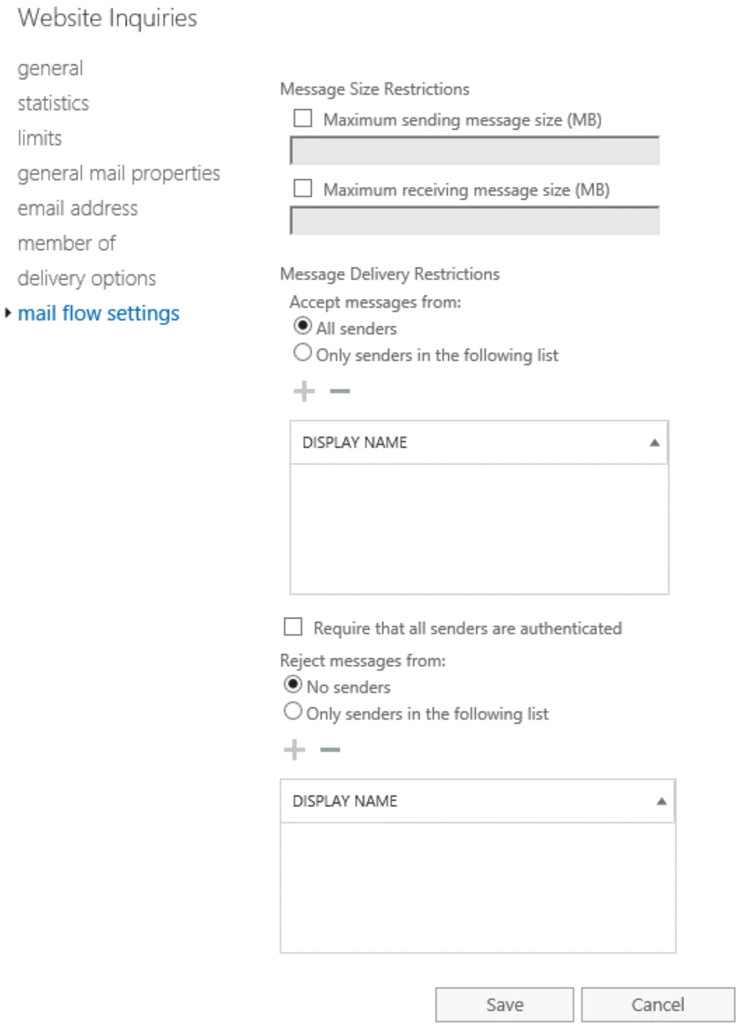
To add an email address, click the **Add** button. On the *New Email Address* dialog specify the *email address type* (in most cases this will be SMTP). Then specify the address in **Email Address** field. Click **Ok**.

To remove an email address, select the address from the list and click **Remove.** Note that only secondary addresses can be removed. If you want to remove an address that is currently the primary address you will first need to assign another primary address.

The **Member Of** tab lists all distribution groups that the public folder is a member. Adding a public folder to a distribution group is useful in archive situations. For example, a distribution group called *All Employees* might be used for sending companywide communications to every employee. Adding a public folder as a member of the distribution group allows a copy of each communication to be delivered to the public folder, creating an archive of all companywide announcements. This member tab is read only so to you will have to add the public folder via the properties of the target distribution group.

The **Delivery Options** tab allows the delegation of *Send as* and *Send on behalf* rights to users and groups. This allows users to send or reply as if they were the public folder. The delivery tab also allows us to set up email forwarding. Any message sent to the public folder can have a copy forwarded elsewhere. The **Deliver message to both forwarding address and mailbox** checkbox determines whether a message is also kept in the public folder.

The **Mail Flow Settings** tab (Figure 8-7) allows us to control how email messages are sent and received by the public folder.



Message Size Restrictions

□ Maximum sending message size (MB)

□ Maximum receiving message size (MB)

Message Delivery Restrictions

Accept messages from:

● All senders

○ Only senders in the following list

DISPLAY NAME

□ Require that all senders are authenticated

Reject messages from:

● No senders

○ Only senders in the following list

DISPLAY NAME

Figure 7: Configuring mail flow settings for a public folder

Message Size Restrictions allows us to control the maximum message size this public folder can send and receive. To set the maximum send size click the **Maximum sending message size (MB)** checkbox and configure a value in megabytes. Likewise, to set a maximum receive size click the **Maximum receiving message size (MB)** checkbox and configure a value in megabytes.

Message Delivery Restrictions provides us with the option of configuring an accept list or a reject list. To configure a list of people who can send to this mailbox select **Only senders in the following list** under *Accept Message From*. Click the **Add** button and pick the users who will be allowed to send messages to this public folder.

Require that all senders are authenticated governs whether external senders can send to this public folder. Selecting this option rejects messages from external senders even if they are defined in the list of allowed senders.

If you would rather block a list of people from sending to this public folder select **Only senders in the following list** under *Reject Messages From*. Click the **Add** button and select the people you want to block.

Whether you pick to build the allow list or the reject list depends on the scope of the block. If you want to reject most start building an allow list. If you want to accept most start building a reject list.

To configure public folder mail options in EMS we need to use the **Set-MailPublicFolder** cmdlet. Note that this is different from the **Set-PublicFolder** cmdlet we mentioned in the previous section. Only the **Set-MailPublicFolder** cmdlet can manage mail attributes.

For example, to modify the alias of *Website Inquiries* our command would look like this.

```
[PS] C:\> Set-MailPublicFolder -Identity "\Website Inquiries" -Alias "Info"
```

To add a second SMTP address to our public folder we can use the **-EmailAddresses** parameter. The email address property is a multi-valued attribute so we need to use the @{Add=""} statement. Without this statement, all existing email addresses would be overwritten.

```
[PS] C:\> Set-MailPublicFolder -Identity "\Website Inquiries" -EmailAddresses
@{Add="smtp:information@space-corp.net"}
```

To remove a secondary email address, we simply switch from add to remove.

```
[PS] C:\> Set-MailPublicFolder -Identity "\Website Inquiries" -EmailAddresses
@{Remove="smtp:information@space-corp.net"}
```

We can check the mail attributes of a public folder by using the **Get-MailPublicFolder** cmdlet.

```
[PS] C:\> Get-MailPublicFolder -Identity "\Website Inquiries" | Format-List
```

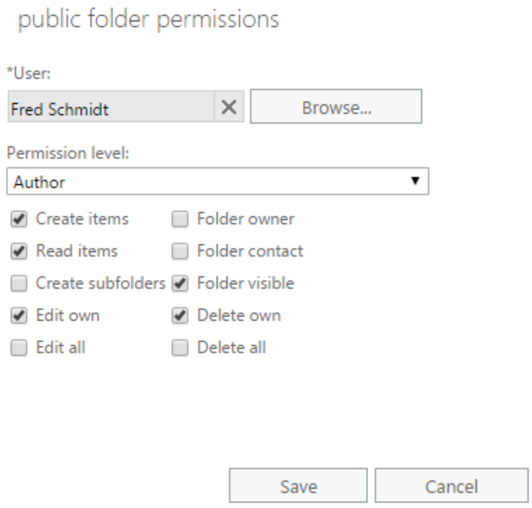
Note: For an extensive list of all available parameters for **Set-MailPublicFolder** cmdlet check the following article https://technet.microsoft.com/en-us/library/bb123707(v=exchg.160).aspx

Managing public folder permissions

Public folder permissions govern what level of access a user has to a public folder.

To configure permissions for a public folder, select the row for that public folder and click the **Manage** link in the task pane. This will launch the public folder permissions dialog. From here we can set permissions for either users or mail-enabled security groups. Note that distribution groups cannot be assigned public folder permissions.

To grant permissions to a user or mail-enabled security group click the **Add** button. This brings up the *Public Folder Permissions* window. Click **Browse** and select a user or group from the list. The **Permission level** drop-down lets you set a predefined access role. An access role is a grouping of common access rights. For example, in Figure 8-8, selecting the *Author* role picks the access rights of read Items, create items, edit own, delete own and folder visible. You can also manually add and remove individual rights by toggling the checkboxes. This will create a custom role. Once you have the desired permissions click **Save**.



public folder permissions

*User:

Fred Schmidt ✕ Browse...

Permission level:

Author ▾

☑ Create items ☐ Folder owner
☑ Read items ☐ Folder contact
☐ Create subfolders ☑ Folder visible
☑ Edit own ☑ Delete own
☐ Edit all ☐ Delete all

Figure 8: Granting public folder permissions to a user

You can add additional users or groups by repeating this process. To edit existing permissions, select the user or group and click the **Edit** button. To remove a user or group's permissions entirely select that object and click **Remove**.

If you want these permissions to propagate to all subfolders check the box **Apply changes to this public folder and all its subfolders**. When you have all the necessary permissions in place click **Save**. The task may take a few seconds to complete. Once completed click **Close** on the confirmation window.

The process to manage permissions for the root folder is almost identical. To access the root permissions, click the **More** button (represented by an ellipsis) and select **Root Permissions**. From the IPM_SUBTREE dialog you can repeat the process of setting permissions as described above.

There are two system groups that cannot be modified through the EAC. These are the *Default* and *Anonymous* groups. We covered the *Anonymous* group briefly in a prior section when we wanted to allow external users to send mail to one of our public folders. *Anonymous* is a system group that represents unauthenticated users. Out of the box this group has no permissions defined. The *Default* group represents authenticated users. This is the fall-back permission when an authenticated user does not have any permissions explicitly defined.

To modify permissions for either of these groups we must use EMS. Let's explore modifying the permissions granted to the *Default* group. First, let's see what permissions are assigned to the folder we want to change.

```
[PS] C:\> Get-PublicFolderClientPermission -Identity "\Calendars"

FolderName           User                 AccessRights
----------           ----                 ------------
Calendars            Default              {Author}
Calendars            Anonymous            {None}
Calendars            Fred Schmidt         {Author}
Calendars            Kendal Ayers         {Editor}
Calendars            Joe Diaz             {Owner}
```

From this view, we can see the default group has *Author* rights. If you wanted to return just the permissions assigned to the default group, you can add **-User** parameter to the previous command. To change the *Author* permission, we actually have to remove the existing permission. For this we use the **Remove-PublicFolderClientPermission** cmdlet.

```
[PS] C:\> Remove-PublicFolderClientPermission -Identity "\Calendars" -User Default -Confirm:$false
```

With the permission removed we can now add our new access role of *Reviewer*. For this we use the **Add-PublicFolderClientPermission** cmdlet.

```
[PS] C:\> Add-PublicFolderClientPermission -User Default -Identity "\Calendars" -AccessRights
Reviewer

FolderName             User                   AccessRights
----------             ----                   ------------
Calendars              Default                {Reviewer}
```

The command returns the new permission assigned to our group. These cmdlets can also be used to add and remove permissions for any other user or group.

In our test environment, the *Calendars* folder is also parent to the *PTO Calendar* and *Corporate Events* folders. If we want to propagate the permissions from calendars into these subfolders we would need to pipe **Get-PublicFolder** into **Add-PublicFolderClientPermission**. For example, to grant Lynn Simmons the *Owner* access role to these three folders our command would look like this.

```
[PS] C:\> Get-PublicFolder -Identity "\Calendars" -Recurse | Add-PublicFolderClient
Permission -User "Lynn Simmons" -AccessRights Owner

FolderName             User                   AccessRights
----------             ----                   ------------
Calendars              Lynn Simmons           {Owner}
Corporate Events       Lynn Simmons           {Owner}
PTO Calendar           Lynn Simmons           {Owner}
```

We can use the same principle to remove permissions in bulk as well. For example, to reverse the permissions we set for Lynn we would issue this command. We add the **-Confirm** parameter so we are not prompted each time a permission is removed.

```
[PS] C:\> Get-PublicFolder -Identity "\Calendars" -Recurse | Remove-PublicFolderClientPermission -
User "Lynn Simmons" -Confirm:$false
```

Piping is also useful when we want to export all permissions to a CSV file. In this command, we omit the **-Identity** parameter from the **Get-PublicFolder** cmdlet. This instructs the command to run from the root versus a subfolder as shown in the previous examples. The **-Recurse** parameter returns all subfolders in the hierarchy which is them piped to **Get-PublicFolderClientPermission** for permissions lookup. The **Select** statement identifies which columns we want to pipe to our **Export-CSV** cmdlet.

```
[PS] C:\> Get-PublicFolder -Recurse -ResultSize Unlimited | Get-PublicFolderClientPermission |
Select FolderName, User, @{Expression={$_.AccessRights};label="Access Rights"} | Export-CSV -Path
"C:\Reports\PublicFolderPermissions.CSV" -NoTypeInformation
```

The following tables detail each access right and access role.

Access Right	***Description***
ReadItems	User can read all items within the folder
CreateItems	User can create items within the folder
EditOwnedItems	User can edit items they have created in the folder
DeleteOwnedItems	User can delete items they have created in the folder
EditAllItems	User can edit any item in the folder
DeleteAllItems	User can delete any item in the folder
CreateSubfolders	User can create subfolders in the folder
FolderOwner	User can view the folder object. User can create subfolders and move the folder. User cannot view, create, edit, or, delete content in that folder.
FolderContact	User is the contact for the public folder
FolderVisible	User can view the folder object. User cannot view or edit content within the folder.

*Table **Error! No text of specified style in document.**-1: Public Folder Access Rights*

Access Role	***Access Rights***
Owner	ReadItems, CreateItems, EditOwnedItems, DeleteOwnedItems, EditAllItems, DeleteAllItems, CreateSubfolders, FolderOwner, FolderContact, FolderVisible.
PublishingEditor	ReadItems, CreateItems, EditOwnedItems, DeleteOwnedItems, EditAllItems, DeleteAllItems, CreateSubfolders, FolderVisible.
Editor	ReadItems, CreateItems, EditOwnedItems, DeleteOwnedItems, EditAllItems, DeleteAllItems, FolderVisible.
PublishingAuthor	ReadItems, CreateItems, EditOwnedItems, DeleteOwnedItems, CreateSubfolders, FolderVisible.
Author	ReadItems, CreateItems, EditOwnedItems, DeleteOwnedItems, FolderVisible.
NonEditingAuthor	ReadItems, CreateItems, FolderVisible
Reviewer	ReadItems, FolderVisible
Contributor	CreateItems, FolderVisible
None	FolderVisible

*Table **Error! No text of specified style in document.**-2: Public Folder Access Roles*

Move a public folder to a different mailbox

Part of public folder administration requires the movement of public folders from one mailbox to another. This is necessary in situations where there is a lack of available space in an existing mailbox, or, there is a need to move popular content for performance purposes to underutilized mailboxes. Depending on the size and popularity of the public folder content it may also be necessary to dedicate an entire database to just a single public folder mailbox. Regardless of how the content is divided the location of the data is transparent to the end user.

When you move a public folder only the content of the folder is moved. The hierarchy remains unchanged. This means that a parent folder could be moved to one mailbox while its subfolders remain in their original mailbox. How long the move takes depends on the size of the folder and the current load of the servers.

To move a public folder, we need to use the **New-PublicFolderMoveRequest** cmdlet. There is no option to do this with the EAC. For example, to move our *Calendars* folder to another mailbox we would issue this command.

```
[PS] C:\> New-PublicFolderMoveRequest -Folders "\Calendars" -TargetMailbox "Public Folder Mailbox 2"
```

You can monitor the progress of this move by running **Get-PublicFolderMoveRequest**.

```
[PS] C:\> Get-PublicFolderMoveRequest

Name                        SourceMailbox               TargetMailbox               Status
----                        -------------               -------------               ------
PublicFolderMove_44d0ea27... space-corp.net/Users/Publi... space-corp.net/Users/Publi... Completed
```

For greater detail, you can pipe the previous command into **Get-PublicFolderMoveRequestStatistics** and optionally add the **-IncludeReport** parameter.

```
[PS] C:\> Get-PublicFolderMoveRequest | Get-PublicFolderMoveRequestStatistics -IncludeReport |
Format-List
```

If you want to move multiple folders at once you can separate each folder path with a comma. For example, if we wanted to move *Corporate Contacts* and its subfolder *Vendors*, while leaving the other subfolders in their original mailbox, we could issue this command.

```
[PS] C:\> New-PublicFolderMoveRequest -Folders "\Corporate Contacts","\Corporate Contacts\Vendors" -
TargetMailbox "Public Folder Mailbox 2"
```

Using the method above to move a large public folder tree that contains hundreds of subfolders would be impractical. While great for moving a handful of folders at once it does not scale well. Luckily there is a script for this exact purpose. The script is called **Move-PublicFolderBranch.ps1** and ships with Exchange in the scripts directory. To complete this process, you will issue the following commands. The first command changes our directory to where the Exchange scripts are located. The second command runs the script with the required variables of the folder root and target public folder mailbox. The folder root identifies which part of the folder tree will be moved with this script. The script creates the move request which can be followed with the **Get-PublicFolderMoveRequest** cmdlet described earlier.

```
[PS] C:\> cd $exscripts
[PS] C:\Program Files\Microsoft\Exchange Server\V15\scripts> .\Move-PublicFolderBranch.ps1 -
FolderRoot "\Corporate Contacts" -TargetPublicFolderMailbox "Public Folder Mailbox 2"

[5/5/2017 5:00:00 PM] Identifying folders that are to be moved to the target mailbox
[5/5/2017 5:00:00 PM] Folders that will be moved as part of this request:

{Corporate Contacts}
{Corporate Contacts, Vendors}
{Corporate Contacts, Customers, A-G}
{Corporate Contacts, Customers, H-N, L Customers}
{Corporate Contacts, Customers, H-N, M Customers}
{Corporate Contacts, Customers, H-N, N Customers}
{Corporate Contacts, Customers, O-U}
{Corporate Contacts, Customers, V-Z}
{Corporate Contacts, Partners}

[5/5/2017 5:00:00 PM] Issuing request to move the public folder branch: \Corporate Contacts
[5/5/2017 5:00:00 PM] RequestName: \PublicFolderMove_44d0ea27-e37f-4a05-8fdc-6001fcfd7a13_f7728513-
b3e9-4211-b349-b4b8a2c42f6c
[5/5/2017 5:00:00 PM] SourceMailbox: space-corp.net/Users/Public Folder Mailbox 1
[5/5/2017 5:00:00 PM] TargetMailbox: space-corp.net/Users/Public Folder Mailbox 2
[5/5/2017 5:00:00 PM] RequestStatus: Queued
[5/5/2017 5:00:00 PM] Use Get-PublicFolderMoveRequest cmdlet to obtain the status of the job.
```

Note: For an extensive list of all available parameters for **New-PublicFolderMoveRequest** cmdlet check the following article https://technet.microsoft.com/en-us/library/jj878081(v=exchg.160).aspx

Only one public folder move request can exist at any one time. To execute a new move request, we need to first delete the existing one. The easiest way to do this is to run the following command.

```
[PS] C:\> Get-PublicFolderMoveRequest | Remove-PublicFolderMoveRequest -Confirm:$false
```

Splitting a public folder mailbox

The maximum size for a public folder mailbox is 100 GB. However, it is recommended to split a public folder mailbox once it reaches 50% capacity. In the last section, we explored how to manually move public folders to a new mailbox. In this section, we explore a script that will automate the split of a public folder mailbox. This script is called **Split-PublicFolderMailbox.ps1** and ships with Exchange. In the following example, we will split *Public Folder Mailbox 2*, retargeting some of its content to *Public Folder Mailbox 1*.

```
[PS] C:\Program Files\Microsoft\Exchange Server\V15\scripts> .\Split-PublicFolderMailbox.ps1

cmdlet Split-PublicFolderMailbox.ps1 at command pipeline position 1
Supply values for the following parameters:
(Type !? for Help.)
SourcePublicFolderMailbox: Public Folder Mailbox 2
TargetPublicFolderMailbox: Public Folder Mailbox 1

[5/5/2017 6:00:00 PM] Determining folders that belong to source mailbox

[5/5/2017 6:00:00 PM] Folders in the public folder hierarchy on the source mailbox
```

```
Path                                                FolderSize     AggregatedSize
\                                                   0              53342316
\Calendars                                          45077912       45077912
\Corporate Contacts                                 0              0
\Corporate Contacts\Vendors                         0              0
\Corporate Contacts\Customers\A-G                   0              8264404
\Corporate Contacts\Customers\H-N\L Customers       0              8264404
\Corporate Contacts\Customers\H-N\M Customers       8264404        8264404
\Corporate Contacts\Customers\H-N\N Customers       0              0
\Corporate Contacts\Customers\O-U                   0              0
\Corporate Contacts\Customers\V-Z                   0              0
\Corporate Contacts\Partners                        0              0

[5/5/2017 6:00:00 PM] Checking if it is possible to split the given source mailbox
[5/5/2017 6:00:00 PM] Source mailbox statistics: Mailbox: Public Folder Mailbox 2 MailboxSize:
52428800 OccupiedSize: 53494207 PublicFoldersOccupiedSize: 53342316

[5/5/2017 6:00:00 PM] Checking if it is possible to move contents to the given target mailbox
[5/5/2017 6:00:00 PM] Target mailbox statistics: Mailbox: Public Folder Mailbox 1 MailboxSize:
52428800 OccupiedSize: 15981311

[5/5/2017 6:00:00 PM] Identifying folders that are to be moved to the target mailbox
[5/5/2017 6:00:00 PM] Possible folder branches for splitting, with total size: 8264404

FolderPath                    FolderSize     SelectedInPass
\Corporate Contacts           8264404        1

[5/5/2017 6:00:00 PM] Selected folder branches for splitting
\Corporate Contacts           8264404

[5/5/2017 6:00:00 PM] Folders that will be moved as part of this request:
{Corporate Contacts}
{Corporate Contacts, Vendors}
{Corporate Contacts, Customers, A-G}
{Corporate Contacts, Customers, H-N, L Customers}
{Corporate Contacts, Customers, H-N, M Customers}
{Corporate Contacts, Customers, H-N, N Customers}
{Corporate Contacts, Customers, O-U}
{Corporate Contacts, Customers, V-Z}
{Corporate Contacts, Partners}

[5/5/2017 6:00:00 PM] Issuing request to split the mailbox: Public Folder Mailbox 2
[5/5/2017 6:00:00 PM] RequestName: \PublicFolderMove_f7728513-b3e9-4211-b349-b4b8a2c42f6c_44d0ea27-
e37f-4a05-8fdc-6001fcfd7a13
[5/5/2017 6:00:00 PM] SourceMailbox: space-corp.net/Users/Public Folder Mailbox 2
[5/5/2017 6:00:00 PM] TargetMailbox: space-corp.net/Users/Public Folder Mailbox 1
[5/5/2017 6:00:00 PM] RequestStatus: Queued
[5/5/2017 6:00:00 PM] Use Get-PublicFolderMoveRequest cmdlet to obtain the status of the job.
```

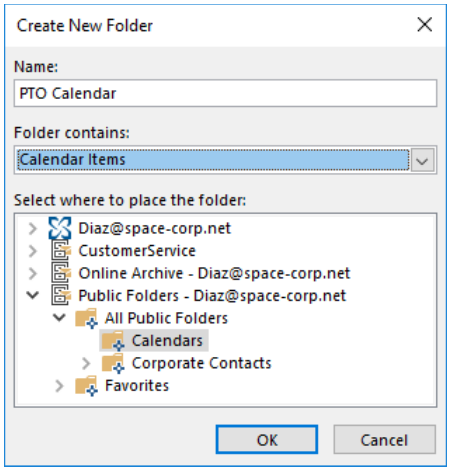
Real World: Exchange ships with dozens of scripts to assist with administrative tasks. Outside of the two scripts we have demonstrated at least nine more scripts exist just for public folders. These scripts can be located in the scripts folder under the Exchange install path. To access via PowerShell type **cd $exscripts**.

Create and manage public folders in Outlook

If a user has the necessary permissions, you can task them with creating public folders by using Outlook. To create a public folder the user will need to be assigned the *Create Subfolders* access right on the parent folder where they are creating the public folder.

To create a public folder using Outlook switch to the **Folders** view and expand **Public Folders > All Public Folders**. Right click on the intended parent folder and select **New Folder** from the context menu. On the *Create new folder* dialog (Figure 8-9) give the folder a name and select the type of content the folder will store. Click **Ok** to create the folder.

In our example, Joe Diaz has created a folder called *PTO Calendar* and set the content type to calendar items. This calendar will be used to track employee vacation time for Space Corp. Joe will automatically be granted the owner access role to this folder, allowing him to further manage it.



Create New Folder

Name: PTO Calendar

Folder contains: Calendar Items

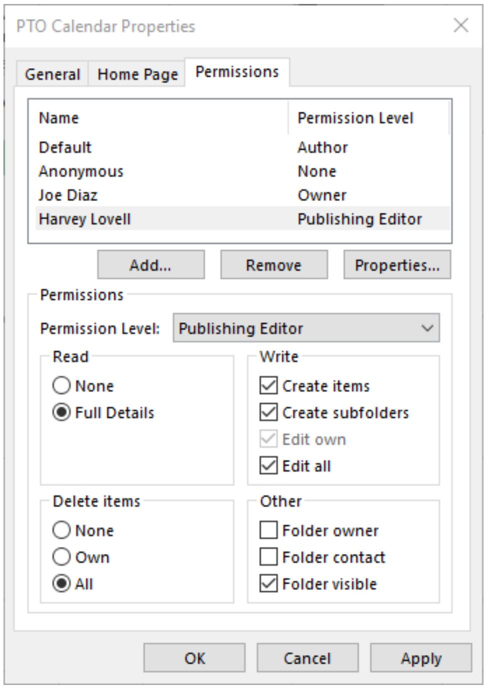
Select where to place the folder:

- Diaz@space-corp.net
- CustomerService
- Online Archive - Diaz@space-corp.net
- Public Folders - Diaz@space-corp.net
  - All Public Folders
    - Calendars
    - Corporate Contacts
  - Favorites

Figure 9: Creating a public folder with Outlook

To manage a public folder right click on the public folder and select **Properties** from the context menu. From the *General* tab the name of the folder can be changed and a description to define the purpose of the folder can be set. Metrics regarding folder size are also available on this tab by clicking the **Folder Size** button.

The *Permissions* tab allows an owner to configure permissions for the folder. By clicking the **Add** button an owner can select a user or group to add to the list and grant them an access role or access rights. For example, Joe has added Harvey Lovell and granted him the *Publishing Editor* access role (Figure 8-10).



PTO Calendar Properties

Name	Permission Level
Default	Author
Anonymous	None
Joe Diaz	Owner
Harvey Lovell	Publishing Editor

Permissions

Permission Level: Publishing Editor

Read
○ None
● Full Details

Write
☑ Create items
☑ Create subfolders
☑ Edit own
☑ Edit all

Delete items
○ None
○ Own
● All

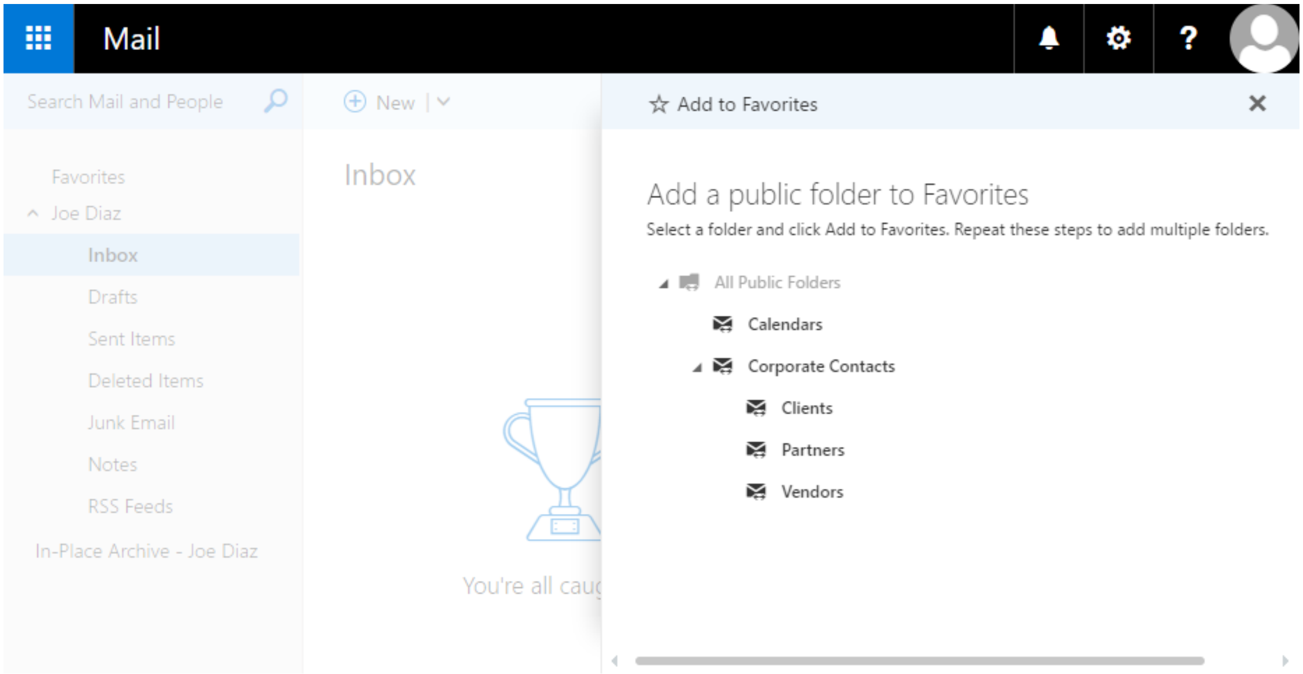
Other
□ Folder owner
□ Folder contact
☑ Folder visible

Figure 10: Managing a public folder's permissions in Outlook

To revoke permissions an owner can select that user and click the **Remove** button. To modify an existing user's permission the owner would select that user from the list and pick a new access role or access right. The **Properties** button displays the selected user's contact card from the global address list. This can help confirm that the correct person is being granted the necessary permissions.

Public Folders in Outlook on the Web

Public folders are a little hidden when it comes to Outlook on the Web. For a user to access a public folder they will need to manually add each one they need. To manually add a public folder in Outlook on the Web right-click on **Favourites** and select **Add public folder to favourites**. On the *Add a public folder to favourites* screen (Figure 8-11) select a public folder and click **Add to favourites**. Repeat this for each additional public folder the user needs.



Add a public folder to Favorites

Select a folder and click Add to Favorites. Repeat these steps to add multiple folders.

Figure 11: Adding public folders in Outlook on the Web

To remove a public folder right click the folder and select **Remove from favourites**.

Delete a public folder

A public folder can be deleted through both the EAC and EMS. To delete a public folder through the EAC select the row for that public folder and click the **Delete** button. You will be warned that this will delete both the folder and any subfolders it may contain. Click **Yes** to confirm or **No** to cancel. The public folder will be removed.

To perform this task in EMS we use the **Remove-PublicFolder** cmdlet. For example, to remove a public folder named *Press Release Packets*, which is a subfolder of *Website Inquiries*, we would issue this command.

```
[PS] C:\> Remove-PublicFolder -Identity "\Website Inquiries\Press Release Packets" -Confirm:$false
```

> **Note:** If the public folder is mail-enabled you will be prompted to mail-disable it before it can be deleted. For this task, use the **Disable-MailPublicFolder** cmdlet. We cover this command in the section titled *Mail-enabling a public folder*.

If you need to delete a folder and all its subfolders you will need to add the parameter **-Recurse** to the command. For example, to delete *Website Inquiries* and its remaining subfolders we would run this command.

```
[PS] C:\> Remove-PublicFolder -Identity "\Website Inquiries” -Recurse -Confirm:$false
```

Delete a public folder mailbox

To delete a public folder mailbox, you first need to move all existing public folders to another mailbox. To determine if the intended mailbox still hosts public folders we can issue the following query.

```
[PS] C:\> Get-PublicFolder -Recurse -ResultSize Unlimited | Where {$_.ContentMailboxName -eq "Public
Folder Mailbox 2"} | Format-Table

Name                                                        Parent Path
----                                                        -----------
Corporate Contacts                                          \
Vendors                                                     \Corporate Contacts
```

In this query, we are performing a lookup of the *ContentMailboxName* field against each public folder in the environment to see if any match the value *Public Folder Mailbox 2*. The result indicates we still have two public folders hosted on *Public Folder Mailbox 2*. We will need to either move or delete these before we can remove the mailbox. For information on how to move a public folder check the section titled *Move a public folder to a different mailbox*.

The second task is to make sure no users have this mailbox set as their default public folder mailbox. To check for this, we can issue the following query.

```
[PS] C:\> Get-Mailbox -ResultSize Unlimited | Where {$_.DefaultPublicFolderMailbox -eq "space-
corp.net/Users/Public Folder Mailbox 2"} | Format-Table Name, DefaultPublicFolderMailbox

Name                                 DefaultPublicFolderMailbox
----                                 --------------------------
Joe Diaz                             space-corp.net/Users/Public Folder Mailbox 2
Fred Schmidt                         space-corp.net/Users/Public Folder Mailbox 2
Kendal Ayers                         space-corp.net/Users/Public Folder Mailbox 2
```

In this query, we are performing a lookup of the *DefaultPublicFolderMailbox* field against each user mailbox in the environment to see if any match the value *Public Folder Mailbox 2*. The result indicates we still have three users configured for *Public Folder Mailbox 2*. We will need to switch those users to a different public folder mailbox. Failure to do this will result in some very poor user experiences in Outlook when accessing public folders. For information on how to change the default public mailbox for a user check the section titled *Public folder hierarchy*.

Once the mailbox is clear of public folders and users we can issue the **Remove-Mailbox** cmdlet. For example, to remove *Public Folder Mailbox 2* our command would look like this.

```
[PS] C:\> Remove-Mailbox -PublicFolder -Identity "Public Folder Mailbox 2" -Confirm:$false
```

We can also use the EAC to delete a public folder mailbox by selecting the mailbox and clicking the **Delete** button. You will be prompted to confirm the deletion. Click **Yes** to confirm or **No** to cancel.

> **Note:** In order to delete the primary hierarchy mailbox all secondary hierarchy mailboxes must first be deleted. In addition, all public folders in the environment must have been removed.

Public vs. Shared vs. Site

Public folders, shared mailboxes and, site mailboxes all share a lot of similarities. It may be difficult to determine which is best for the job when it comes to collaboration.

With site mailboxes, you receive the added benefit of attaching a SharePoint document library to a mailbox. This makes site mailboxes the perfect candidate for project teams. It allows a team to store documents and emails conversations in a central repository, all accessible by the Outlook client. Where this presents a challenge is that it does require a SharePoint 2013 infrastructure. For those without SharePoint and no plans to implement this will quickly rule out site mailboxes.

Another possibility is that a site mailbox may just be overkill for the task at hand. For example, if all the business requires is a shared calendar to track employee vacations a document library might be unnecessary. On the other hand, some businesses may see a site mailbox as a perfect fit, using the document library to store policies and procedures regarding the submission of time off requests.

The other issue is that site mailboxes have been deprecated in Office 365. With all the innovation and development focused on the cloud and then trickled down on-prem it is unlikely we will see any new features for site mailboxes.

As we have seen public folders are organized in a hierarchy of folders, much like a local drive or network share. This can conceivably make resources easier to find through a folder tree. Although the counter to that argument is AutoMap. With AutoMap, shared mailboxes can be published directly to a user without their intervention. This will certainly cut down on support calls.

For on-demand archived content, like we mentioned in our *All Employee* distribution group example, having this content mapped permanently with AutoMap could be unnecessary. It is possible to turn AutoMap off but then it becomes a race to see which method is faster to access.

While a public folder can be mail-enabled if a team needs a full mailbox experience then a shared mailbox is going to be the better choice. For example, a customer service department may receive support inquiries to a customer service shared mailbox. With the full mailbox experience that department can track all inbound mail, all sent mail, and process mail into folders either manually or with rules. Not to mention the options for contacts, calendars and tasks. On the flipside if a team simply needs a single folder experience the public folder might be the better choice.

Microsoft positions the shared mailbox for team collaboration. In contrast public folders are often attributed to organization level tasks.

What you pick depends on the current need of the business. You certainly don’t need to be committed to any one solution. There is nothing wrong in using all three if it serves a business goal.

Summary

In this chapter, we took a look public folders. We discussed how to perform common administrative tasks through the Exchange Admin Center (EAC) and the Exchange Management Shell (EMS). Tasks that included the creation, deletion and management of public folders and public folder mailboxes.

In the next chapter, we explore addressing including; email address policies, address lists and, procedures for mergers and rebranding.