Microsoft Exchange Server 2013 Inside Out: Mailbox and High Availability

Tony Redmond

ISBN: 978-0-7356-7858-3

First printing: September, 2013

To ensure the ongoing accuracy of this book and its companion content, we've reviewed and confirmed the errors listed below. If you find a new error, we hope you'll report it to us on our website: www.microsoftpressstore.com/contact-us/errata.

Page	Location	Description	Date corrected
51, 58, 62, 63	Throughout Chapter 2	Setup.com is referenced six times in Chapter 2. Each instance should be replaced with Setup.exe.	
109	"Too many objects" sidebar	The number of objects returned by EMS is 10,000 not 1,000. The text for the sidebar should be as follows: By default, EMS returns up to 10,000 objects in response to cmdlets. (The value in Exchange 2007 is 5,000.) Therefore, if you run a cmdlet such as Get-Mailbox, Exchange will return up to 10,000 mailboxes if they are available. If you work in a small Exchange organization that supports fewer than 10,000 mailboxes, you don't need to worry too much about the number of objects you have to deal with because PowerShell will likely return relatively few objects, and things usually progress quickly. However, it's a different situation in large organizations, where you have to pay attention to the filters you specify to retrieve data or override the default limit for returned objects by specifying the ResultSize parameter for cmdlets. For example, to let EMS return as many mailboxes as it can find, you could use a command like this: Get-Mailbox –ResultSize Unlimited This command will work, but it will be very slow because EMS has to read every single mailbox in the organization. Think about how long this might take to execute in an organization that supports more than 300,000 mailboxes. In these situations it's always better to specify a filter to restrict the number of objects EMS looks for and returns.	

Page	Location	Description	Date corrected
215	Paragraph after Figure 5-16, first sentence	Reads: Mailboxes that use the quotas inherited from the database settings have their UseDatabaseQuotaDefaults property set to $False, whereas those that have individual quota settings have the property set to $True. Should read: Mailboxes that use the quotas inherited from the database settings have their UseDatabaseQuotaDefaults property set to $True, whereas those that have individual quota settings have the property set to $False.	
485	First paragraph under bulleted list	The paragraph should read: Transaction logs are not copied to servers that host database copies when block mode replication is in use because the passive servers are already building and replaying their own copy of logs. However, as soon as the Replication service determines that block mode is not appropriate (for instance, the replay queue has begun to grow), it reverts to file mode replication and starts to copy transaction logs again. The advantage of block mode is that transactions are dispatched immediately when they are available to the log buffer of the active database. Transaction data arrive to the servers that hold passive databases as quickly as the server hosting the active database can dispatch data across the network.	
658	Last paragraph, penultimate line	Reads: ...such as the Calendar, Contacts, and Tasks,… Should read: ...such as the Calendar and Tasks,…	
661	First Inside Out sidebar	Should read: INSIDE OUT Some items are timeless Items in some folders tend to be more timeless than general-purpose messages, so you should think carefully through the potential consequences when you create retention policy tags for folders. For example, people usually want to keep important project documents for a long time, so it might be best to create tags that enable users to mark these items not to expire. To do this, they should create a personal tag with a Never retention period, which indicates to Exchange that the item should be held indefinitely and neither deleted nor archived by MFA.	

Page	Location	Description	Date corrected
770	Inside Out sidebar	The sample code contains formatting problems. The complete sidebar should read: INSIDE OUT Knowing which public folder mailbox holds the writable hierarchy As you can see from Figure 12-2, many public folder mailboxes can be active in an organization, but only one holds the writable copy of the public folder hierarchy. EAC shows you which public folder mailbox contains the primary hierarchy, but you can also find out and discover how EAC knows which mailbox holds the writable copy of the public folder hierarchy by issuing this command: Get-OrganizationConfig | Select RootPublicFolderMailbox However, this command only returns the globally unique identifier (GUID) of the public folder mailbox, which is not really useful to a human. The better approach is to run this command: Get-Mailbox -PublicFolder | Where {$_.IsRootPublicFolderMailbox –eq $True} | Select Name, ExchangeGuid This returns a list of all public folder mailboxes and then applies a filter to extract the mailbox that contains the writable hierarchy.	