Part V

Sharing Content

Chapter 15

Sharing Content on the Power BI Service (PowerBI.com)

In This Chapter

- **The Power BI Service and Data Refresh**
- **The Power BI Service and Sharing**
- **The Power BI Service and Row-Level Security**

We have spent a number of chapters learning how to create great content in Power BI. The purpose of this content is to allow users to explore the data, discover insights, and make data-driven decisions. In order for any of that to happen, we need to find a way to share this content with the users.

We can use two platforms to share content with users. The first is the Power BI Service (PowerBI.com). The other is the Power BI Report Server. The Power BI Service is part of Microsoft's Azure platform and, therefore, as we saw in Chapter 5, resides in the cloud. Power BI Report Server is designed to be an on-premises installation.

The remaining chapters of this book are dedicated to these two Power BI report-sharing environments. In this chapter, we explore the mechanisms for sharing content within the Power BI Service. In Chapter 16, we explore the operation of the on-premises Power BI Report Server.

THE POWER BI SERVICE AND DATA REFRESH

In Chapter 5, we explored the operation of PowerBI.com. You learned how to navigate and work with the various content created and published by Power BI Desktop. You created a dashboard from the published report content.

We won't rehash the use of PowerBI.com here. If you need a refresher, please reread Chapter 5. Instead, most of this chapter will focus on sharing content. You will learn how to create content packs and apps in order to deliver Power BI content to others. Before we get to that, we need to cover one other very important process. How do we make updated data coming from our on-premises sources available to data models deployed in the cloud?

The On-premises Data Gateway

Eventually, you will want to load new data into the data models loaded on PowerBI.com. This is not a problem if the data was taken from a public source somewhere in the cloud. It is also pretty straightforward if the data source resides on Microsoft Azure.

Things get a bit more complicated if the data source is on one of your organization's private servers. This is where the

On-premises Data Gateway comes in. The On-premises Data Gateway provides a secure conduit from your on-premises data sources to data consumers in Azure like Power BI data models. Once in place, the On-premises Data Gateway provides a safe path for your Power BI data models to refresh their data either on a scheduled basis or on demand. The On-premises Data Gateway will also support Live Connect access to on-premises SQL Server Analysis Services instances as well as Direct Query access to SQL Server relational instances and to SQL Server Analysis Services.

The On-premises Data Gateway operates on an organization's server infrastructure and is usually maintained by the IT department. It is set up to be shared by a number of applications at that organization needing to move data to the Microsoft cloud. A personal version, the On-premises Data Gateway (personal mode), is also available.

The personal version operates from a workstation rather than a server. As the name implies, it is intended for personal, not organization-wide, use. The On-premises Data Gateway (personal mode) only supports data refresh for Power BI models using Import mode. It does not support Direct Query or Live Connect.

The On-premises Data Gateway Architecture

The On-premises Data Gateway operates by making a secure, outbound HTTP connection to the Microsoft Azure environment. The gateway operates using the same communication path a browser does when making a secure connection to a website. Because this connection uses the

standard secure HTTP protocol, the On-premises Data Gateway does not require any special ports or security holes to be opened in your organization's firewall.

Figure 15-1 shows the operation of the On-premises Data Gateway. The gateway can be used to support Power BI operating in Import, Direct Query, or Live Connect mode. In fact, a single installation of the On-premises Data Gateway can handle data requests for all three modes of operation.

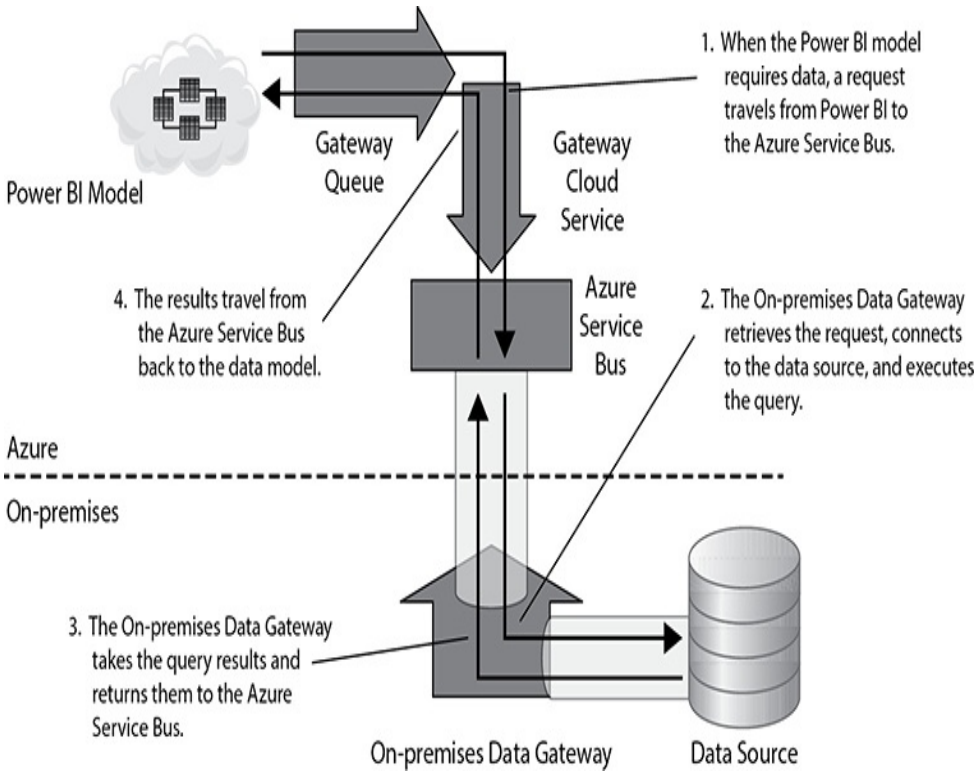

Figure 15-1 *The On-premises Data Gateway architecture*

Import Mode Data Model Refresh and Direct Query

When the On-premises Data Gateway is used to refresh a model in Import mode or is used to provide data to a model in Direct Query mode, a request containing the following is sent from the Power BI model to the gateway:

- Data query to be executed
- Data source information
- Data source connection credentials (encrypted)

When the On-premises Data Gateway receives this request, it decrypts the data source connection credentials. It then uses the data source information along with the credentials to connect to the appropriate data source. A single installation of the On-premises Data Gateway may connect to multiple on-premises data sources based on the data source information in the request. The gateway installation just has to have a path through the on-premises infrastructure to connect to each specified data source.

Once the connection is created, the On-premises Data Gateway executes the data query on that data source. The result set from that query execution is returned to Azure. Azure, in turn, gets the result set back to the Power BI model.

Live Connect

When the On-premises Data Gateway is used to provide data to Power BI in Live Connect mode, a request containing the following is sent from Power BI to the gateway:

- Data query to be executed
- Data source information
- Data source connection credentials (encrypted)
- User principal name of the user currently connected to PowerBI.com

When the On-premises Data Gateway receives this request, it decrypts the data source connection credentials. It then uses

the data source information along with the credentials to connect to the appropriate data source.

Once the connection is created, the On-premises Data Gateway uses the user principal name to create the security context of that user. The data query is then executed on the data source in that security context. In this manner, any security restrictions set on that user within the data model being queried will be in effect. The users who explore data through Live Connect mode on PowerBI.com can only retrieve data they have rights to see. As before, the result set from that query execution is returned to Azure, and Azure gets the result set back to Power BI.

NOTE

The following descriptions are intended to serve as a guide should you need to install and configure an On-premises Data Gateway. They are not meant to be a step-by-step exercise. If you do wish to try this process with the “Max Min Sales Information.pbix” file, you will need to restore several SQL Server databases on a default instance of SQL Server 2017 (or later) on the same computer where the On-premises Data Gateway is installed. The database backup files for this process can be found in the “Databases to Use with Data Gateway Example” folder in the supporting materials for this book.

Installing the On-premises Data Gateway

The On-premises Data Gateway software is available as a free download from Microsoft. To download, sign in to PowerBI.com, click the Download button (down arrow icon)

in the Power BI toolbar, and then select Data Gateway. This is shown in Figure 15-2 and will open the On-premises Data Gateway browser page in a new tab.

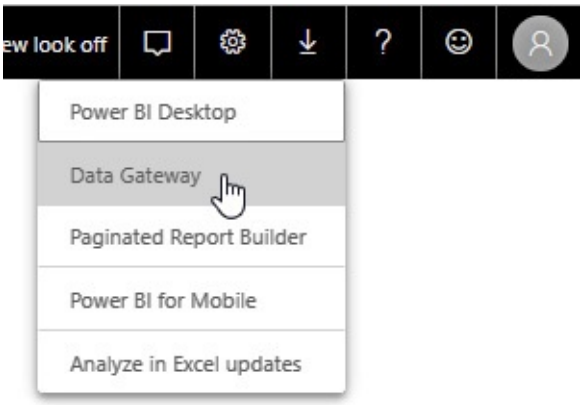

Figure 15-2 *The Download Data Gateway menu option*

Click the Download Gateway button to download a copy of the installation executable. Execute this file to begin the gateway installation process. Follow the prompts during the installation process. As part of the install process, you can select whether this is a regular installation or a personal mode installation, as shown in Figure 15-3.

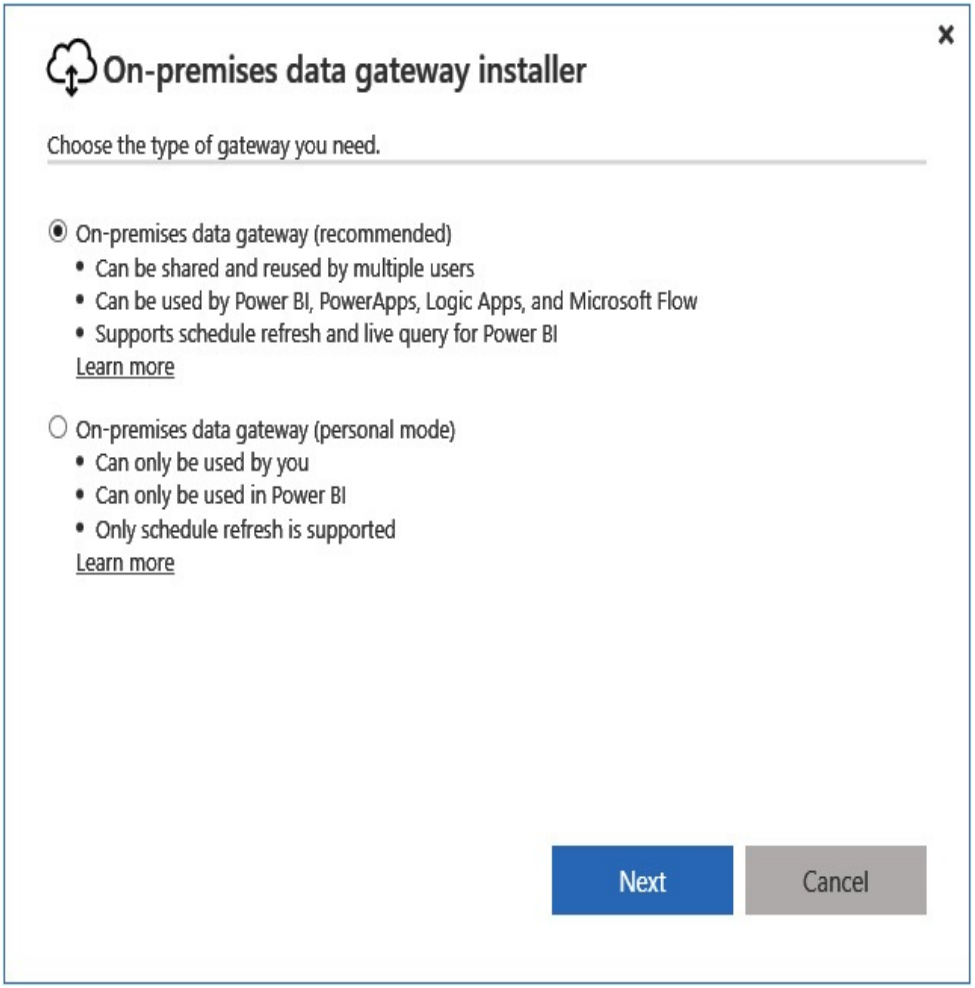

Figure 15-3 *Choosing the On-premises Data Gateway type*

The personal mode On-premises Data Gateway can be installed on your personal computer—hence the name. Just keep in mind this gateway will only be active when your computer is running. Also keep in mind this program generates some overhead as it looks for data requests.

The regular version of the gateway does not need to be installed on the same server where the data source is running. However, this can be a good idea if that server has the available capacity. This will reduce the network overhead of data-request data moving between the data source and the

gateway software.

If you installed the regular On-premises Data Gateway, the install program will display the screen shown in Figure 15-4 when complete. This screen shows the Azure programs, including Power BI, that are ready to use the gateway.

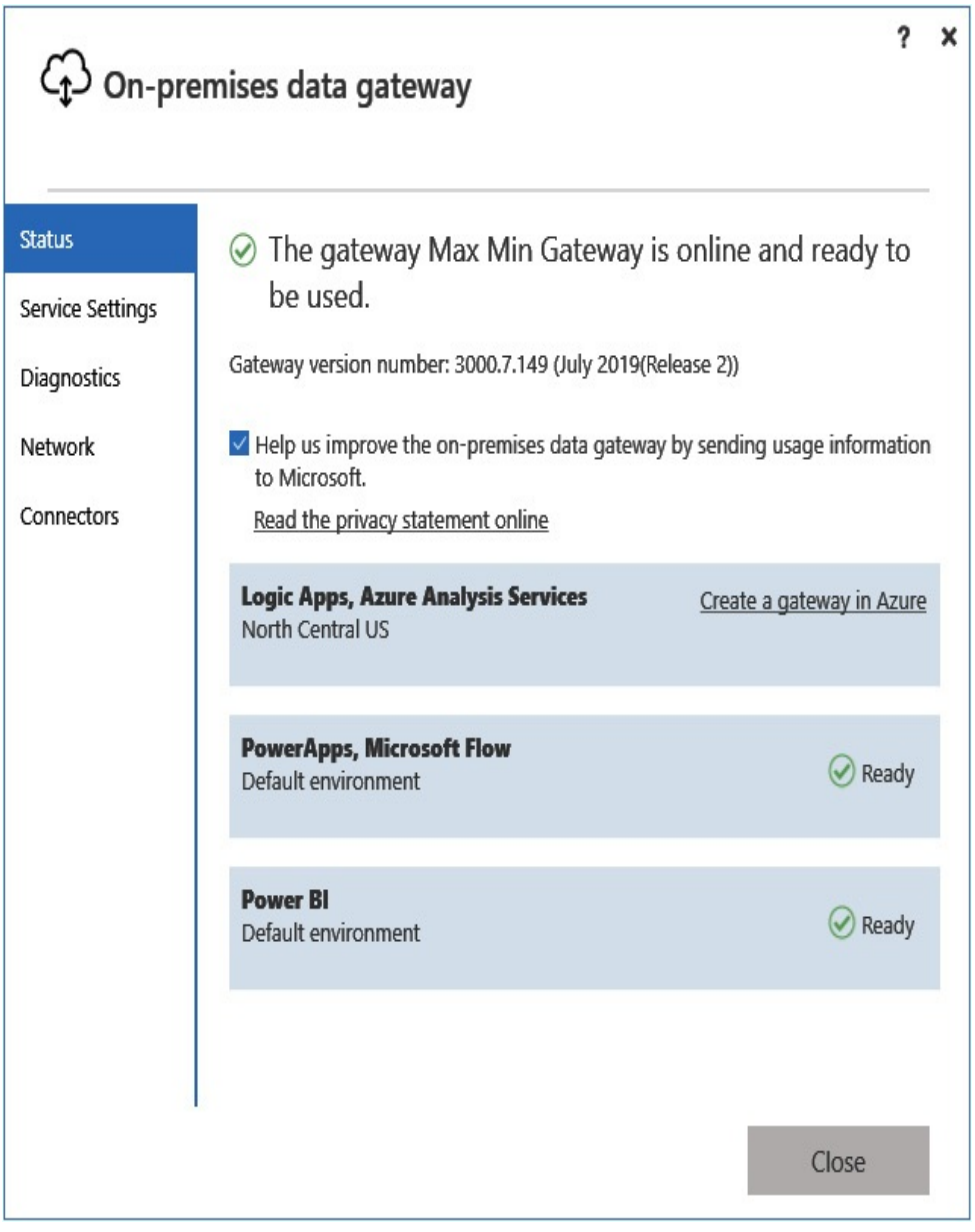

Figure 15-4 *The completed On-premises Data Gateway installation*

Managing the On-premises Data Gateway

Once you have installed the On-premises Data Gateway, you can manage the gateway from PowerBI.com. This includes ensuring the gateway is up and running and configuring your PowerBI.com content to use the gateway for its data needs. The On-premises Data Gateway and the On-premises Data Gateway (personal mode) are managed a bit differently. This section covers the On-premises Data Gateway processes. The "Managing the On-premises Data Gateway (personal mode)" section covers the personal mode processes.

If you installed the regular On-premises Data Gateway, you can use the Manage Gateway pages of PowerBI.com to view and modify gateway settings. These pages will not function if you installed the On-premises Data Gateway (personal mode) version, as noted in the "Managing the On-premises Data Gateway (personal mode)" section of this chapter. To view the Manage Gateway pages, click the Settings button (cog icon) in the Power BI toolbar and then select "Manage gateways." This is shown in Figure 15-5.

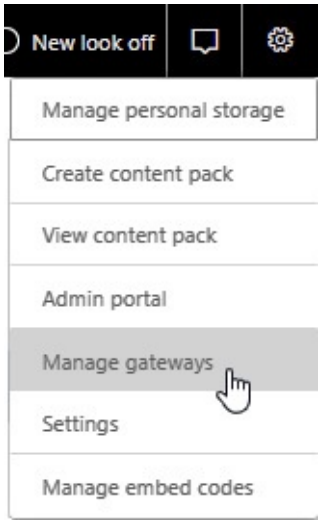

Figure 15-5 *The Manage gateways menu option*

Selecting “Manage gateways” displays the Gateway Cluster Settings tab of the Gateways page, as shown in Figure 15-6. Even if you only have a single gateway installed, the tab will be titled “Gateway Cluster Settings.” In addition to setting data identifying this gateway, you can set the following options:

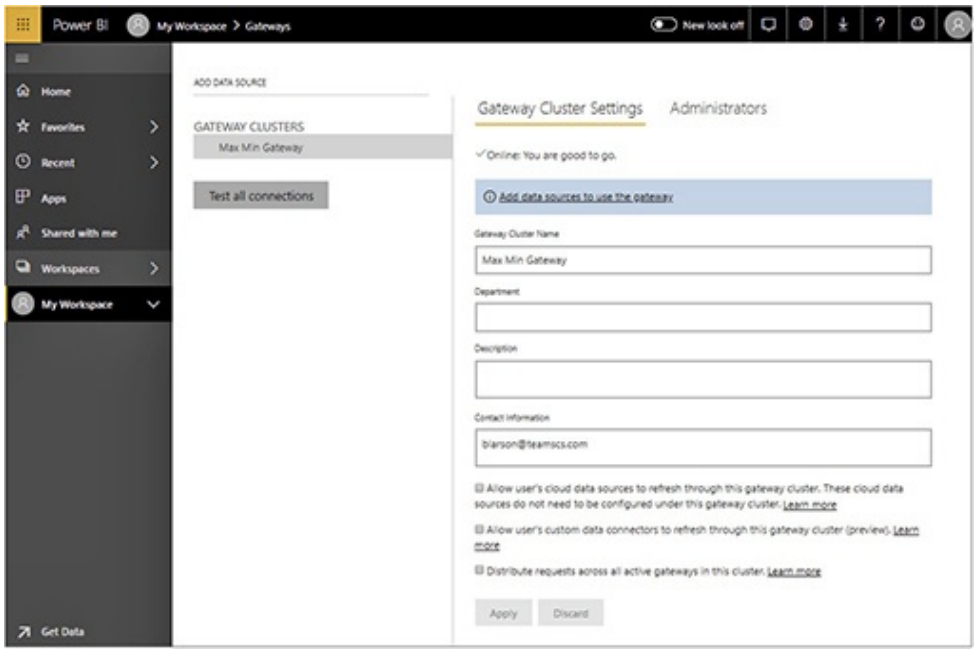

Figure 15-6 *The Gateway Cluster Settings tab*

- **Allow user’s cloud data sources to refresh through this gateway cluster** When this option is enabled, data from on-premises sources can be combined with data from cloud sources in the same query.
- **Allow user’s custom data connectors to refresh through this gateway cluster** If you have developed custom data connectors using the Data Connector SDK (software development kit), you must enable this option to allow

those connectors to use the gateway.

- **Distribute requests across all active gateways in this cluster** When this option is enabled and there are multiple gateways configured in this cluster, enabling this option will allow for load balancing across all active gateways.

Clicking Administrators at the top of the page takes you to the Administrators tab of the Gateways page, as shown in Figure 15-7. This tab enables you to define who has rights to administer each gateway. The person who installed the gateway is automatically added as an administrator of the gateway.

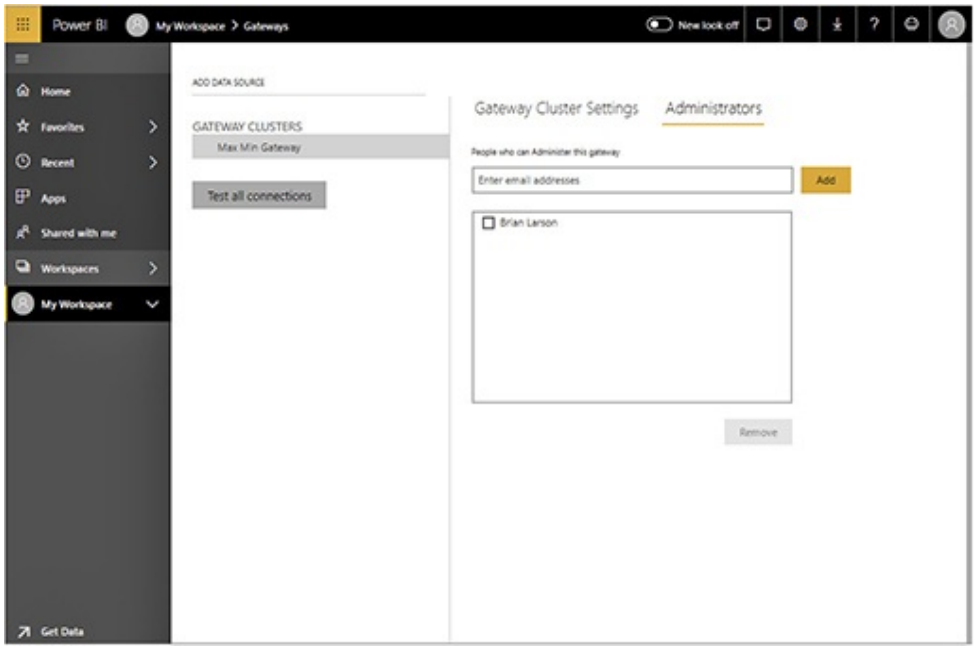

Figure 15-7 *The Gateway Administrators page*

Adding a Data Source to an On-premises Data Gateway

In order for the On-premises Data Gateway to be used by your

published Power BI content, you must define data sources for the gateway. Again, this does not apply to an On-premises Data Gateway (personal mode) installation. See the "Managing the On-premises Data Gateway (personal mode)" section of this chapter for the personal mode process.

To add a data source, click the ADD DATA SOURCE link, as shown in Figure 15-8. This can be done from either the Gateway Cluster Settings tab or the Administrators tab. Clicking this link will take you to the Data Source Settings tab. This tab enables you to define a data source, including information such as type of connection, server name, database name, and Windows credentials. The Data Source Settings tab is shown in Figure 15-9.

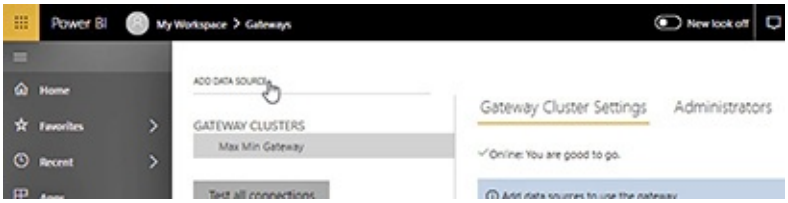

Figure 15-8 *Adding a gateway data source*

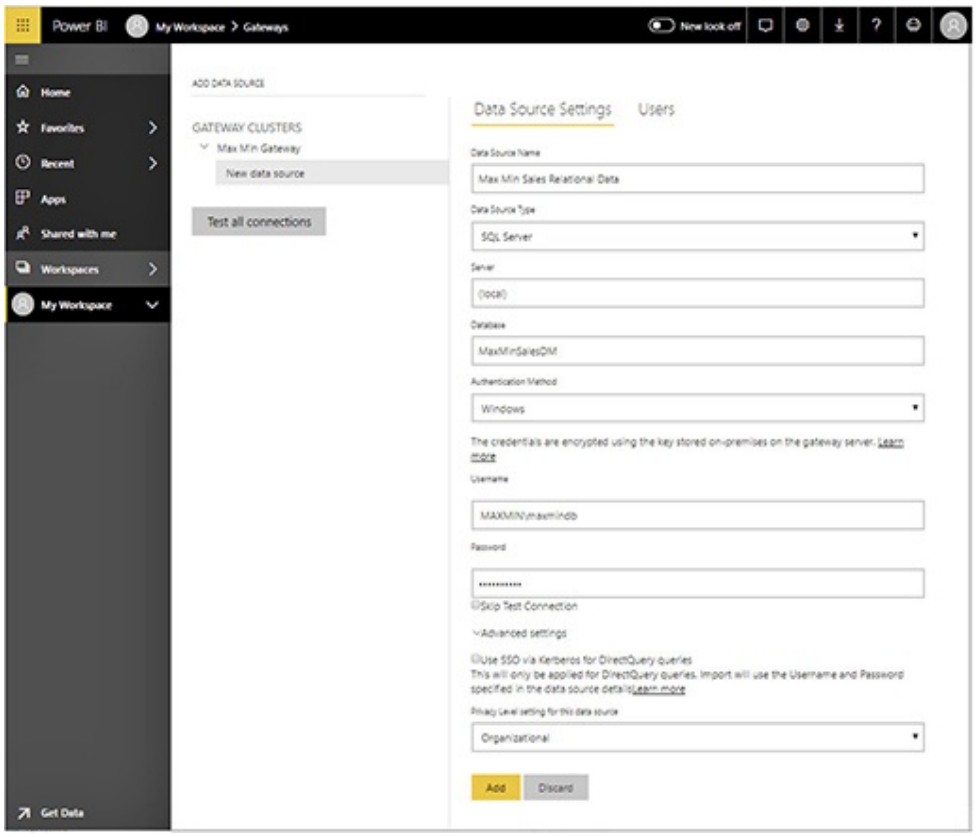

Figure 15-9 *The Data Source Settings tab*

NOTE

The server name and database name used in a gateway data source definition must exactly match the server name and database name in the Power BI dataset in order for that dataset to use the gateway to access on-premises data.

Once the data source is defined, the Users tab is used to specify those people who can use this gateway data source. Users in this list may publish Power BI content and then configure that content to access on-premises data through this data source. The Users tab is shown in Figure 15-10.

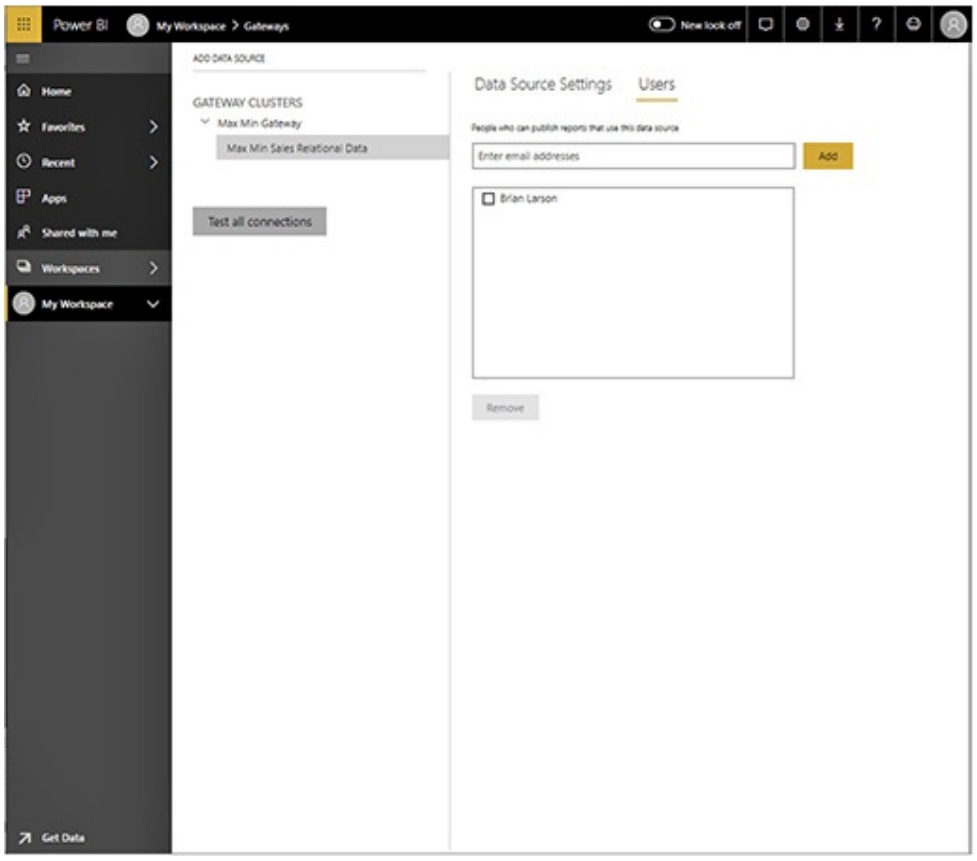

Figure 15-10 *The Users tab*

Once a data source is defined for a gateway, you can return to that data source's configuration information by expanding the gateway entry, as shown in Figure 15-11. To delete a data source, hover over the far-right end of the gray box surrounding the data source name and click the "Open menu" button (…) that appears to the right. Use the REMOVE option to delete the data source, as shown in Figure 15-12.

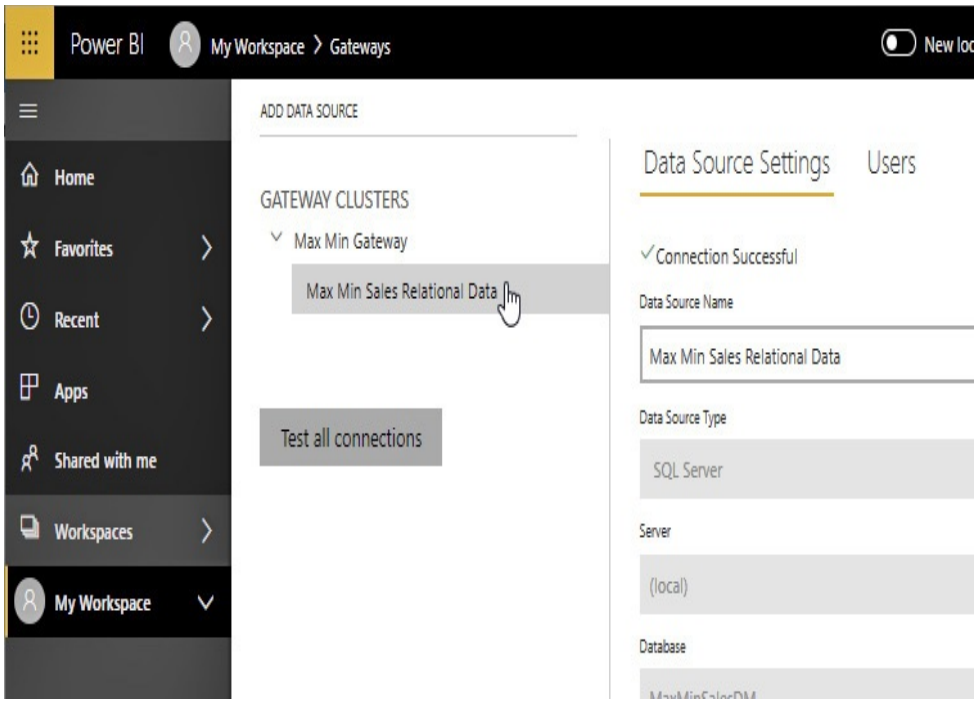

Figure 15-11 *Returning to a data source definition*

Figure 15-12 *Removing a data source*

Configuring a Gateway Data Source for a Dataset

The final step in this process is to configure Power BI datasets to use the gateway data sources. To accomplish this, click the Settings button (the cog icon) in the Power BI toolbar and then select Settings from the menu. This is shown in Figure 15-13.

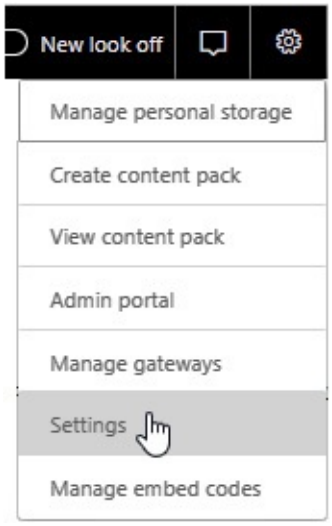

Figure 15-13 *The Settings menu option*

This will take you to the General tab of the Settings page. Click Datasets to navigate to the Datasets tab. Once on the Datasets tab, you can click a dataset to view the property settings for that dataset. Finally, expand the "Gateway connection" item, as shown in Figure 15-14.

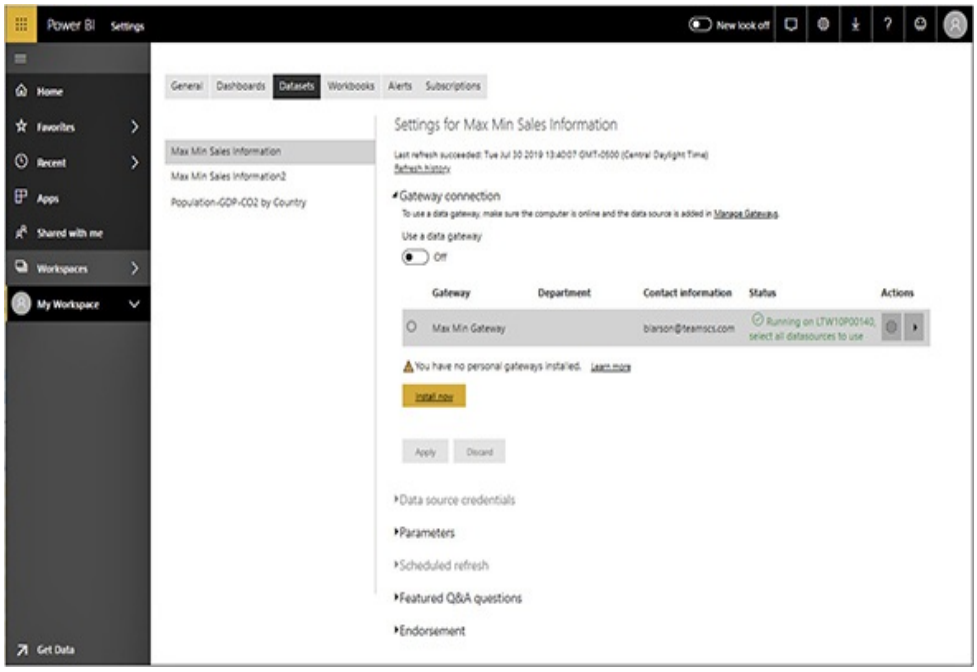

Figure 15-14 *The Datasets tab of the Settings page with no*

gateway data source in use

To configure a gateway data source for a dataset, click the arrow at the far end of the gateway entry, as shown in Figure 15-15. Use the “Maps to” dropdown list to select the appropriate gateway data source for each data source in the Power BI dataset, as shown in Figure 15-16. (Selecting the gateway data sources will automatically turn on the “Use a data gateway” slider.) Once all of the dataset data sources are configured with gateway data sources, click Apply.

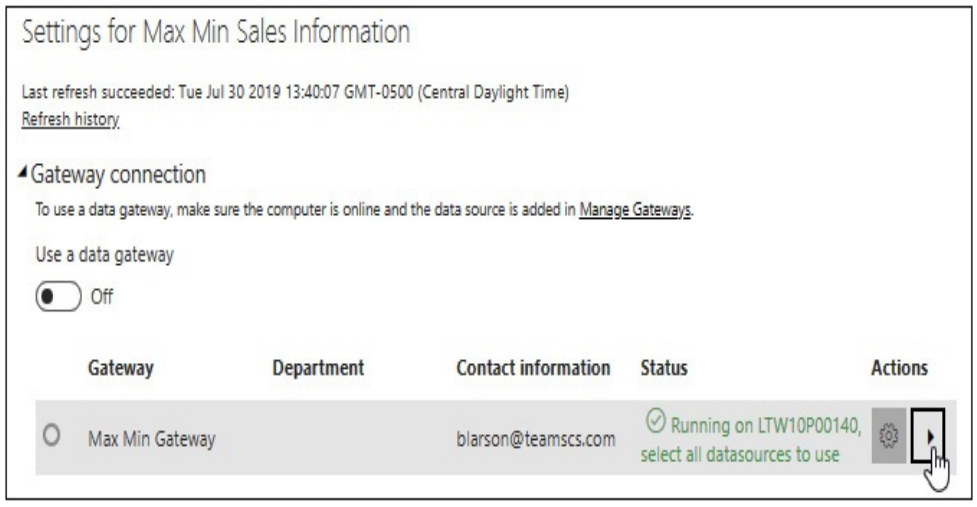

Figure 15-15 *Expanding a gateway to configure for a dataset*

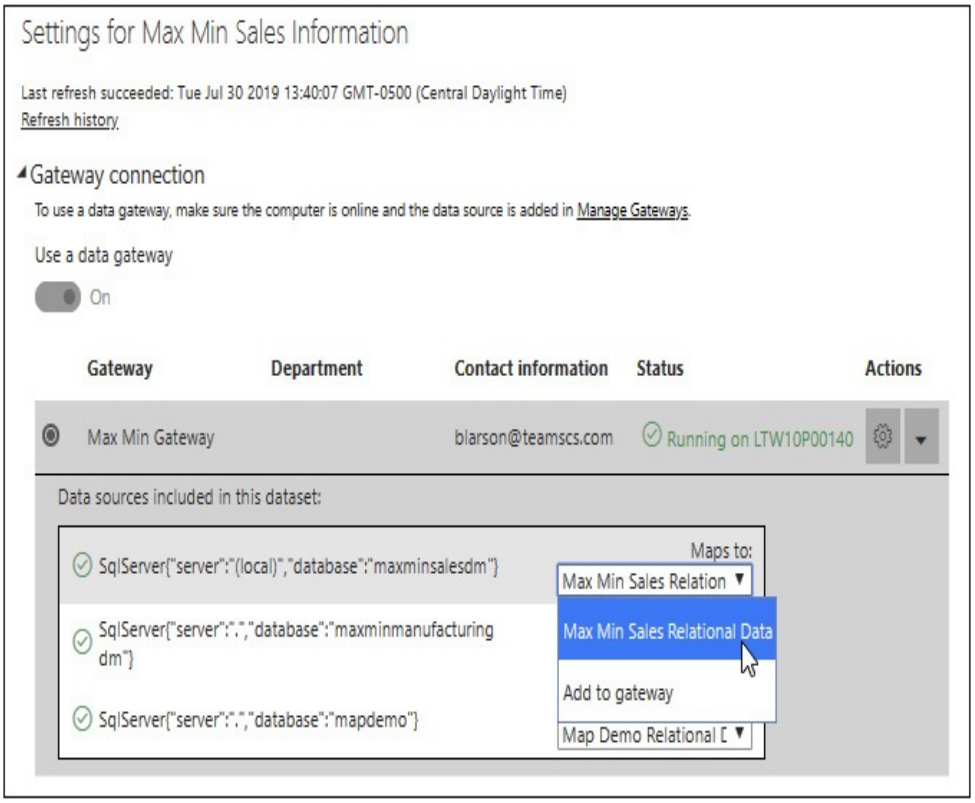

Figure 15-16 *Selecting the gateway data source for the dataset*

Managing the On-premises Data Gateway (personal mode)

Once you have installed the On-premises Data Gateway (personal mode), you can configure Power BI datasets to use the gateway in PowerBI.com. To accomplish this, click the Settings button (the cog icon) in the Power BI toolbar and then select Settings. This is shown earlier in Figure 15-13. This will take you to the General tab of the Settings page. Click Datasets to navigate to the Datasets tab.

NOTE

The personal mode gateway does not require us to configure gateway data sources.

Once on the Datasets tab, you can click a dataset to view the property settings for that dataset. Finally, expand the "Gateway connection" item as shown in Figure 15-17. Figure 15-17 shows a dataset that can use the available personal gateway. Simply select the radio button next to Personal Gateway and click Apply. (Clicking the Personal Gateway radio button will automatically turn on the "Use a data gateway" slider.) If required, edit the data source credentials to provide a valid user name and password.

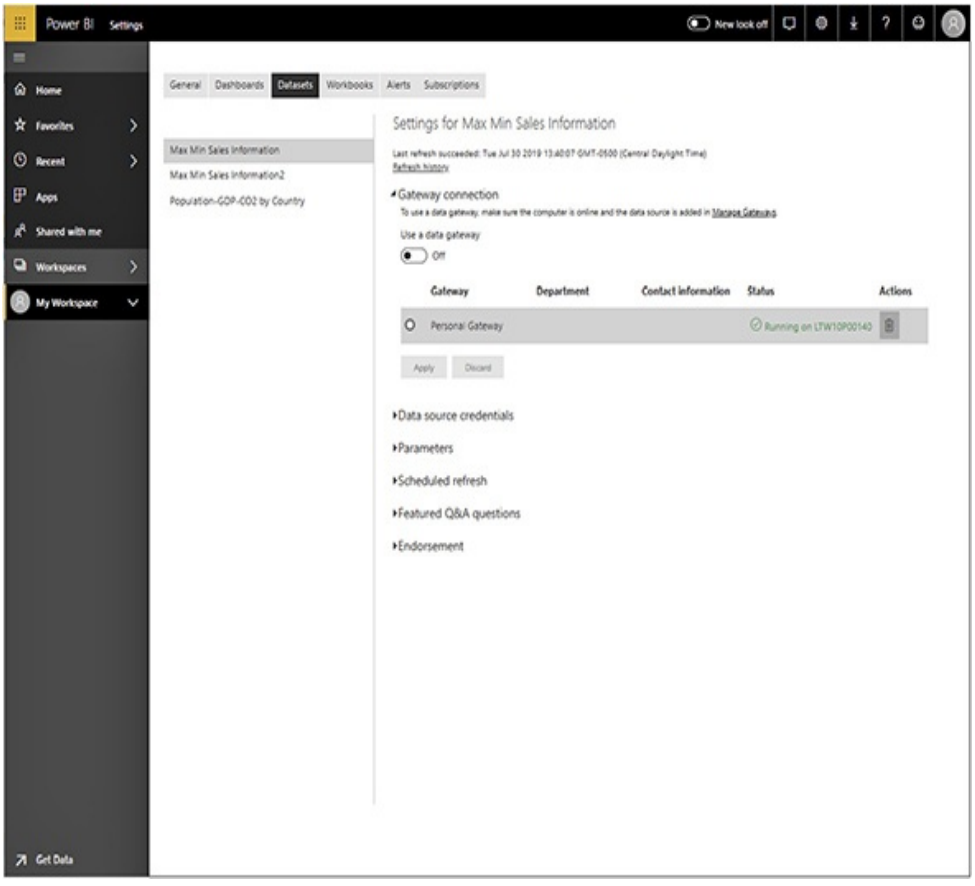

Figure 15-17 *A dataset eligible to use the personal gateway*

Setting a Dataset for Scheduled Refresh

Once you have an import dataset configured to use the On-premises Data Gateway, you may set up that dataset to perform a scheduled data refresh. This is done on the Datasets tab of the Settings page. You can reach the Datasets tab of the Settings page through the Settings menu, as we did in the previous section, or you can click the "Schedule refresh" button on the entry for that dataset in the Datasets area of the workspace, as shown in Figure 15-18.

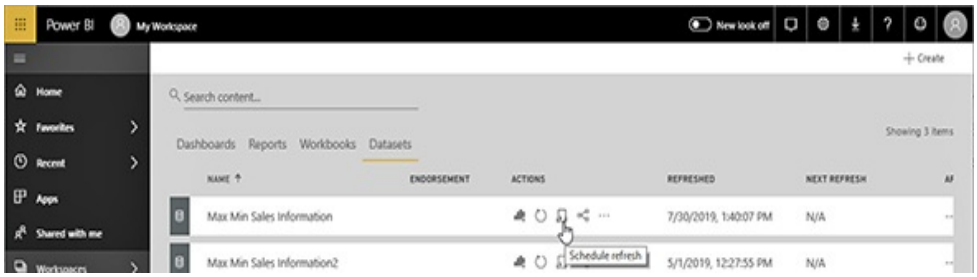

Figure 15-18 *The "Schedule refresh" button for a dataset*

Once on the Dataset tab, expand the "Scheduled refresh" item. Change the "Keep your data up to date" slider to On. Enter the required information on the refresh schedule you would like to use for this dataset and provide an email to be contacted if there is an issue during data refresh. Once completed, the schedule refresh configuration should appear similar to Figure 15-19. Click Apply to begin using this scheduled refresh.

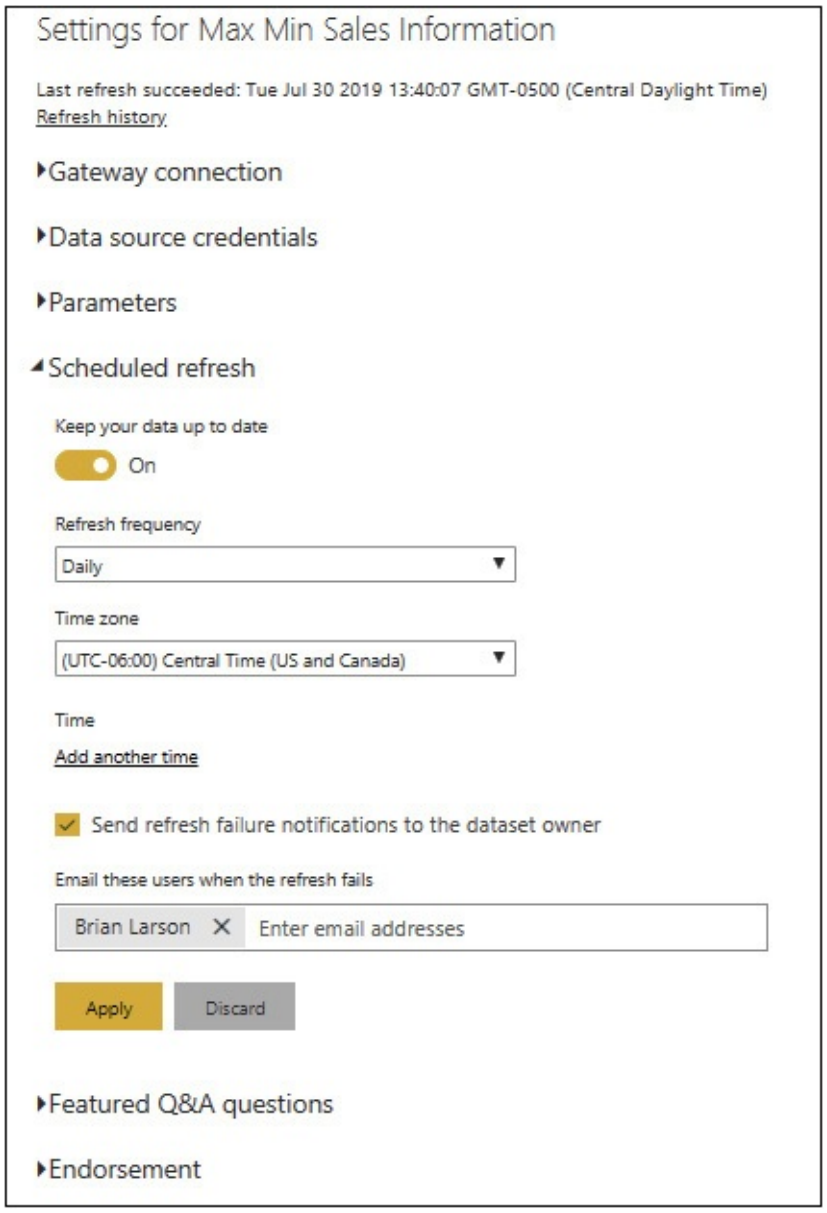

Figure 15-19 *Configuring a scheduled refresh*

THE POWER BI SERVICE AND SHARING

Publishing content to PowerBI.com puts a copy of our Power BI reports in an environment where we can easily access our own content through a browser-based interface. However, our

Power BI content works pretty well for us in Power BI Desktop. Just having the Power BI content available through a browser may not be a good enough reason to publish content.

What we probably want to do when we publish Power BI content is to make it available to others. In Chapter 5, we saw there were several ways to do that. First, we can share reports and dashboards with others using the Share button. We can also create content packs and apps to deliver our content to others.

Using Share

Report sharing and dashboard sharing are accomplished using the Share button in the entry for a report and for a dashboard, respectively. This process was covered in the "Share" section under "Reports" in Chapter 5. We won't rehash that topic here. For more information, refer back to that section in Chapter 5.

Using Content Packs

Content packs allow us to package together dashboards, reports, and datasets. The package can then be made available for others to install. When a user installs a content pack, its dashboards, reports, and datasets are simply mixed in with the other dashboards, reports, and datasets that happen to be in that workspace.

Content packs can be made available to a select group of people, specified by email address, or to your entire organization. They are given a title, a description, and, optionally, an icon to differentiate them from other content packs that might be available to a user.

We'll walk through the process of creating a content pack,

installing that content pack, and finally deleting that content pack.

NOTE

Organizational content packs require a Power BI Pro license for both the person creating the content pack and the person installing the content pack.

Creating a Content Pack

1. As part of this exercise, we will use the “Analytics Icon.jpg” image file. Locate this file in the supporting materials download and copy it to a location in your file system.
2. Open a browser and navigate to PowerBI.com.
3. Click “Sign in” at the top of the page.
4. Complete the sign-in process.
5. Select My Workspace on the left side of the web page. My Workspace opens with the Dashboards tab active.
6. Click the “Data Analysis with Microsoft Power BI” dashboard. This dashboard opens in the browser.
7. Click the Settings button (the cog icon) in the Power BI toolbar and select “Create content pack” from the menu. This is shown in Figure 15-20.

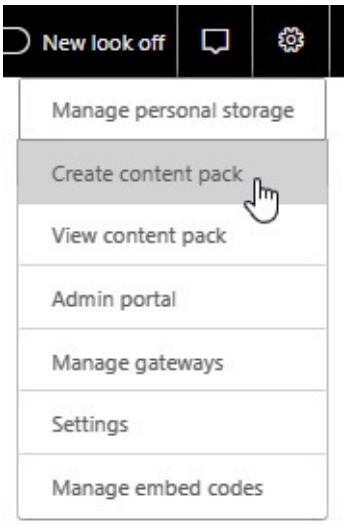

Figure 15-20 *The "Create content pack" menu option*

8. Select the "My entire organization" option.
9. For Title, enter **Data Analysis with Power BI Test Content Pack**.
10. For Description, enter **This is a test content pack**.
11. In the phrase "Upload an image or company logo," click the word "Upload." This opens a file selection dialog box.
12. Navigate to the folder where you copied the "Analytics Icon.jpg" image file.
13. Select the "Analytics Icon.jpg" image file and click Open.
14. In the "Select items to publish" area, click the check box for the "Data Analysis with Microsoft Power BI" dashboard. (The dashboard name may be shortened with an ellipsis at the end.) The reports that this dashboard depends on are automatically selected. The datasets that those reports depend on are also automatically selected.

The content pack creation page should appear, as shown in Figure 15-21.

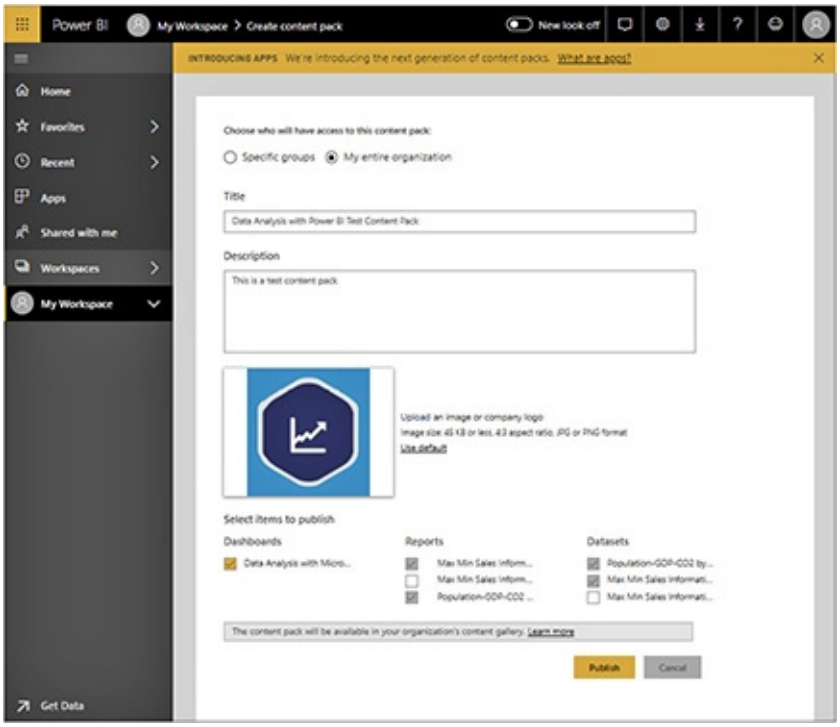

Figure 15-21 *Creating a content pack*

15. Click Publish. The content pack is created.

Installing a Content Pack

For a moment, we will play the role of another user who wants to get the dashboard and reports in our content pack. Our fictional user will find the content pack we just created and install its content.

1. Click Get Data in the lower-left corner of the Power BI page in your browser. The Get Data page appears.
2. Click the Organizational Content Packs link, as shown in Figure 15-22.

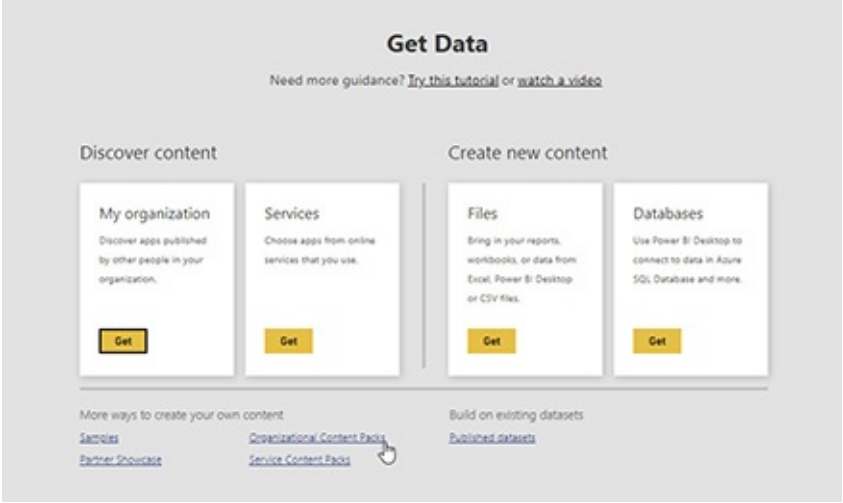

Figure 15-22 *The Organizational Content Packs link on the Get Data page*

3. Click the "Get it now" button for the "Data Analysis with Power BI Test Content Pack," as shown in Figure 15-23.

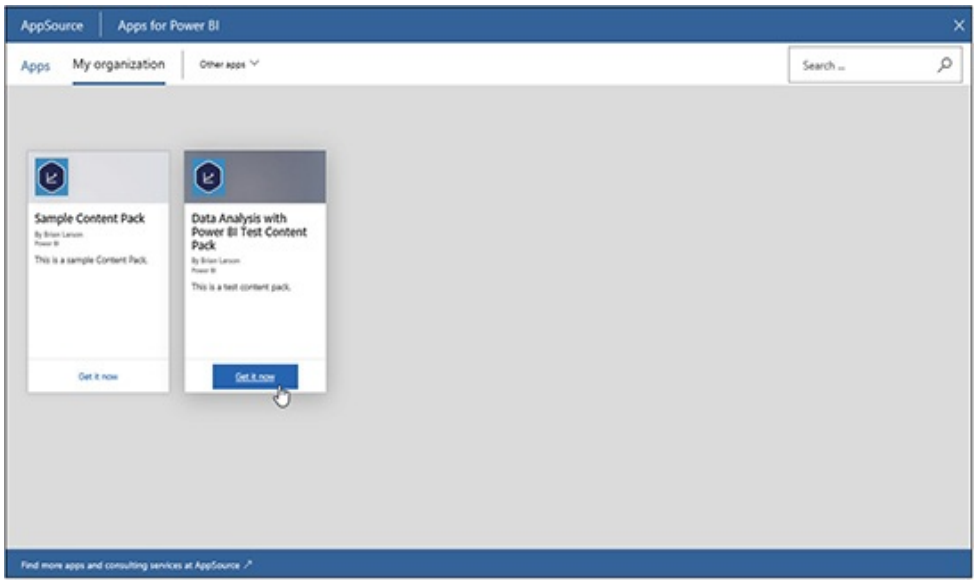

Figure 15-23 *Installing a content pack*

A second copy of the Data Analysis with Microsoft Power BI dashboard is created in My Workspace, as shown in Figure

15-24. Note the second copy of the dashboard does not have the buttons for sharing or for controlling settings. Our fictional user does not have the right to share or control the settings of our dashboard. If the content of My Workspace is expanded on the left, you can also see that duplicates of the reports were installed.

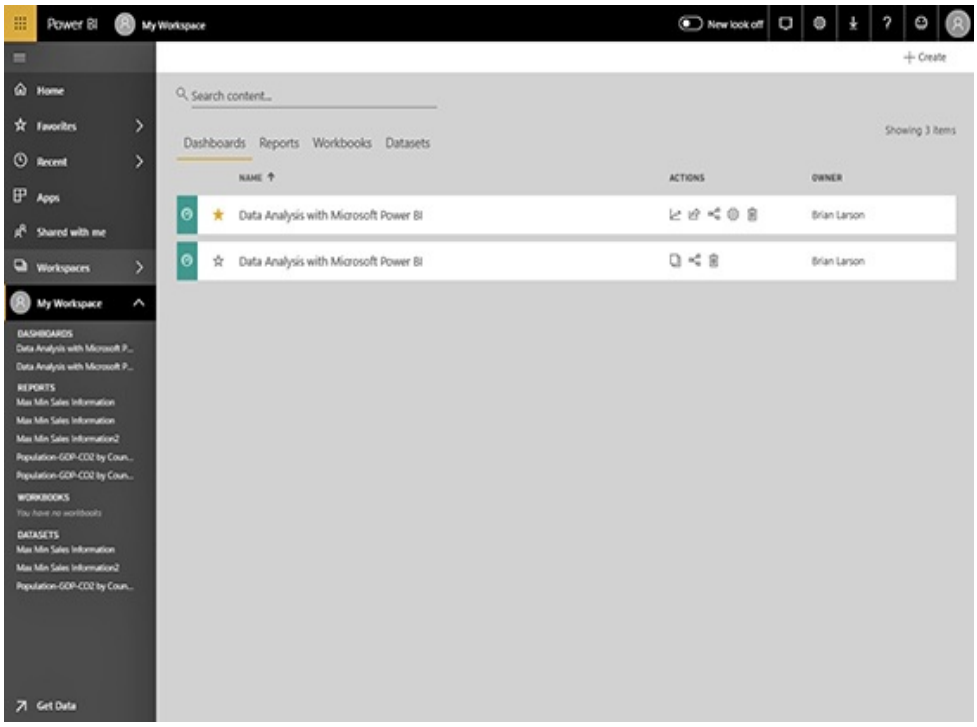

Figure 15-24 *The results of the content pack installation*

Deleting a Content Pack

1. Click the Settings button (the cog icon) in the Power BI toolbar and select “View content pack” from the menu. The “View content pack” page appears, as shown in Figure 15-25.

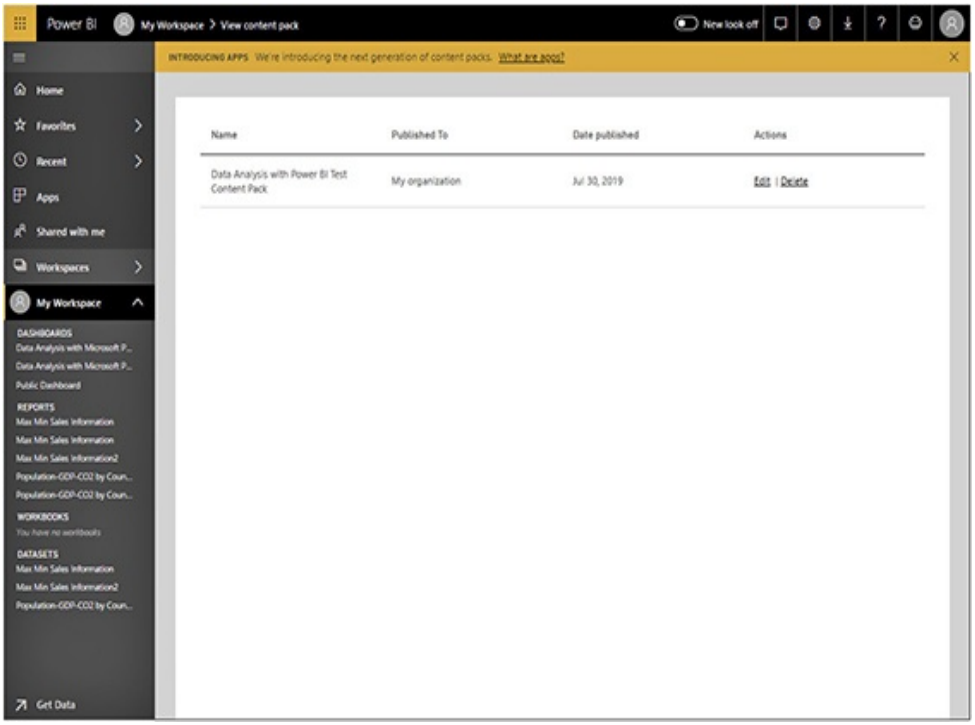

Figure 15-25 *The "View content pack" page*

2. Click Delete on the right side of the Data Analysis with Power BI Content Pack entry. The "Delete content pack" dialog box appears.
3. Click Delete. Note the duplicate items created from the content pack have been removed from My Workspace on the left side of the page. When the content pack is deleted, its installed items are also deleted.

Using Apps

Currently, apps are preferred over content packs for sharing content. Like content packs, apps package together dashboards, reports, and datasets for sharing with others. Unlike content packs, installed apps end up in their own area within PowerBI.com. Each app has its own name and icon. The dashboards, reports, and datasets from each app remain

separate, and the app content is accessed through the app icon.

We'll try creating an app, installing that app, and finally deleting that app.

Creating an App

1. Click Workspaces on the left side of the PowerBI.com page.
2. Click "Create a workspace" at the bottom of the workspaces listing, as shown in Figure 15-26. The "Create a workspace" dialog box appears.

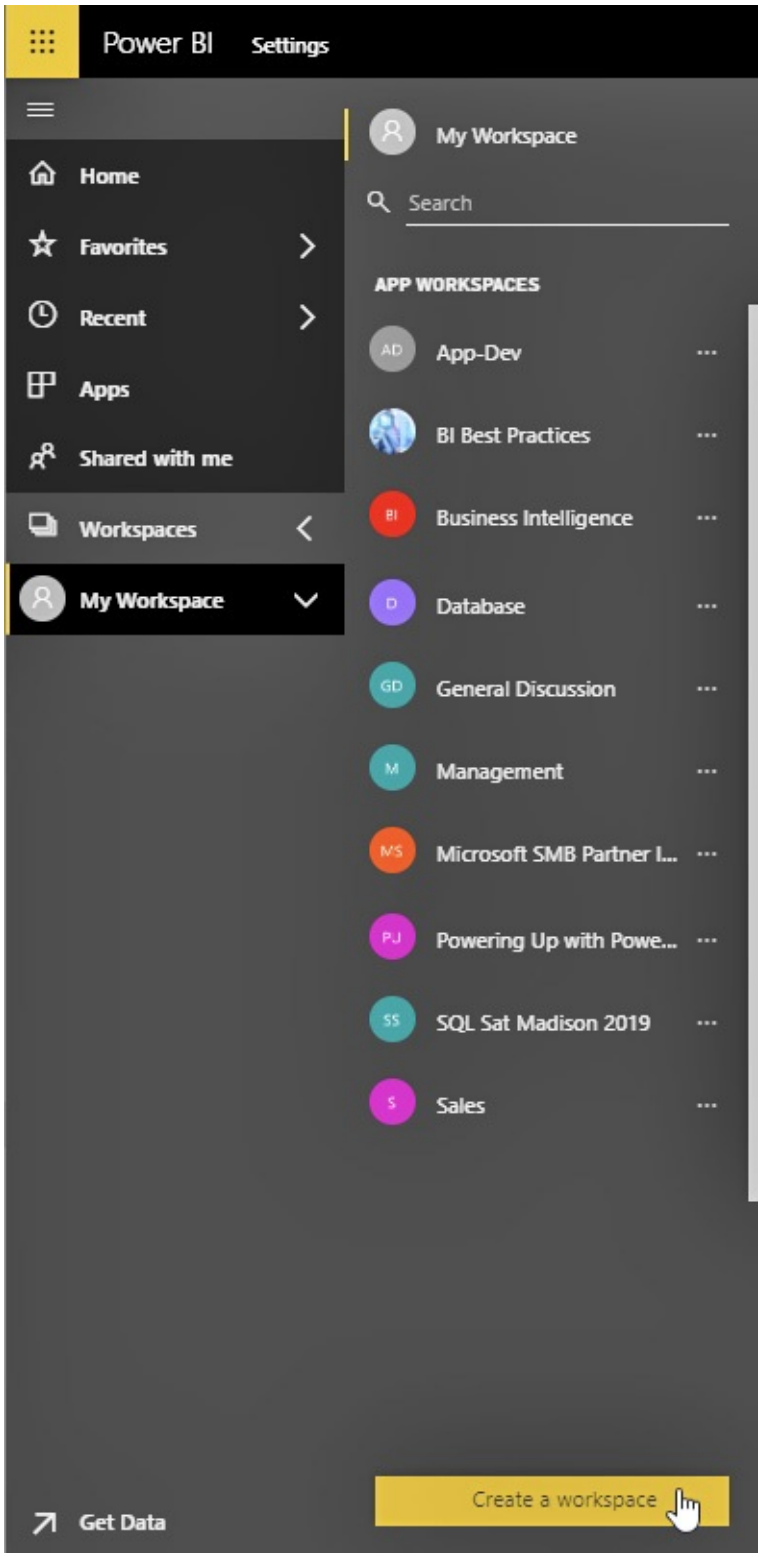

Figure 15-26 *"Create a workspace" button*

3. For Workspace name, enter **Data Analysis with Power BI Test App**.
4. For Description, enter **This is for a test app.** The completed dialog box should appear as shown in Figure 15-27.

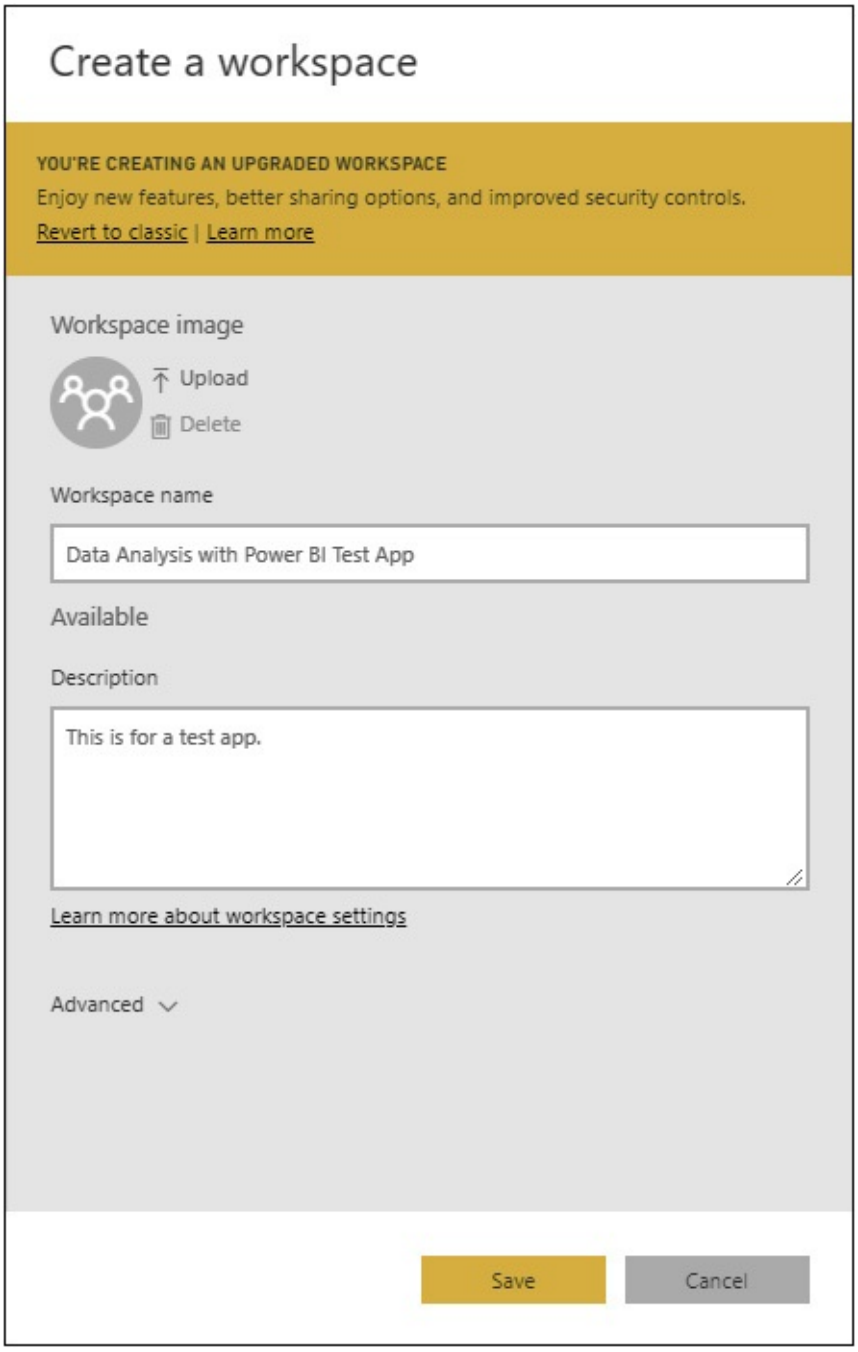

Figure 15-27 *Creating a new workspace*

5. Click Save. The new workspace appears in the workspaces listing.
6. Open the "Max Min Sales Information.pbix" file in Power BI Desktop. We will deploy content from here into our new workspace.
7. On the Home tab of the ribbon, click Publish. The Publish to Power BI dialog box appears.
8. Select the "Data Analysis with Power BI Test App" workspace, as shown in Figure 15-28.

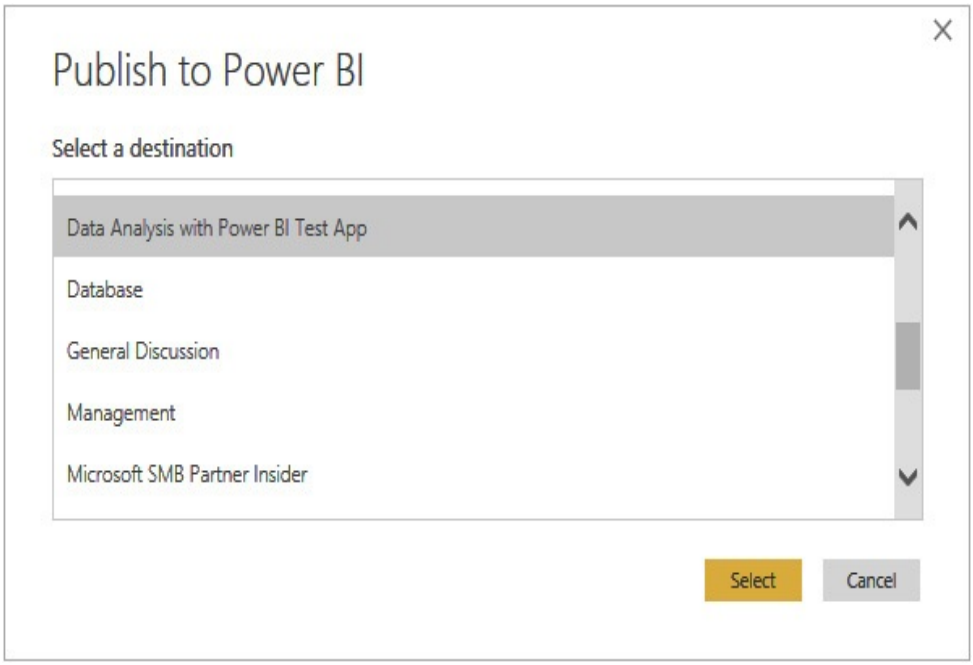

Figure 15-28 *Publishing to the new workspace*

9. Click Select. The content is deployed to the "Data Analysis with Power BI Test App" workspace on PowerBI.com.
10. When the Success! dialog box appears, click "Got it."

11. Close Power BI Desktop.
12. In the browser, select the “Data Analysis with Power BI Test App” workspace in the workspaces listing. This workspace is opened.
13. Select the Reports tab.
14. Click the “Max Min Sales Information” report to open it.
15. On the “Sales by Sales Person” tab, hover over the “Sales in Dollars by Sales Person” chart and click the “Pin visual” button (pushpin icon) at the top of the chart, as shown in Figure 15-29. The “Pin to dashboard” dialog box appears.

Figure 15-29 *Pinning a chart to a dashboard*

16. Ensure that the “New dashboard” radio button is selected and enter **App Dashboard** for the Dashboard name.
17. Click Pin.

18. Select the “Sales Units by Promotion” tab.
19. Hover over the “Sales in Units by Promotion” treemap and click the Pin visual button.
20. Ensure App Dashboard is selected and click Pin. Close any popup messages that may appear.
21. Return to the Data Analysis with Power BI Test App workspace using the breadcrumbs at the top of the page, as shown in Figure 15-30.

Figure 15-30 *Returning to the workspace page*

22. On the Dashboards tab, ensure the INCLUDED IN APP toggle is set for App Dashboard, as shown in Figure 15-31. The report that this dashboard comes from will also be included in the app.

Figure 15-31 *Including content in the app*

23. Click the Publish app button at the top of the screen, as shown in Figure 15-32. The App Setup page appears.

Figure 15-32 *Starting the app-publishing process*

24. For Description, type **This is a test app.**
25. Select an app theme color using the color picker. The Setup page appears, as shown in Figure 15-33.

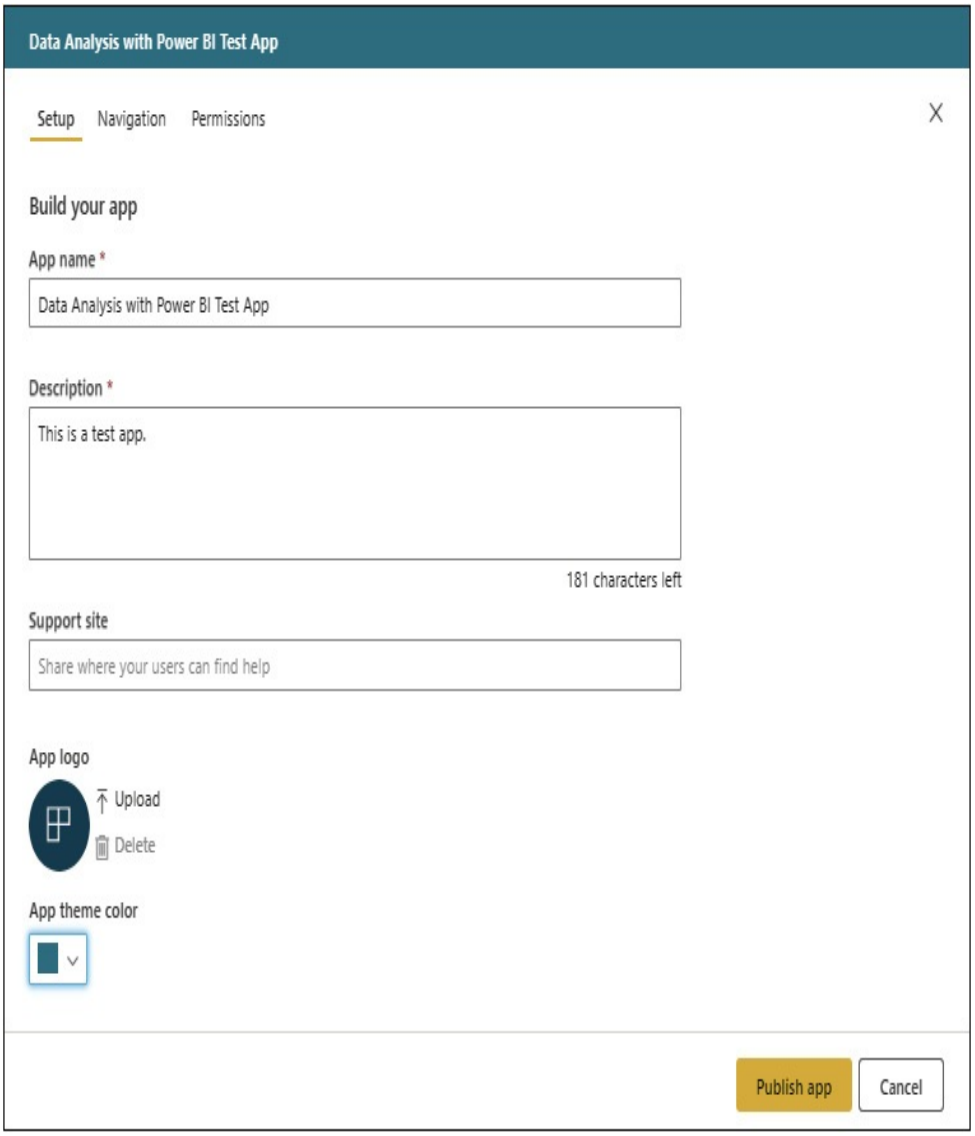

Figure 15-33 *The Setup page*

26. Select Navigation at the top of the page. Using the Navigation area on the left, you can select which content will be the landing location for the app. Leave the App Dashboard at the top of the list. The Navigation page appears, as shown in Figure 15-34.

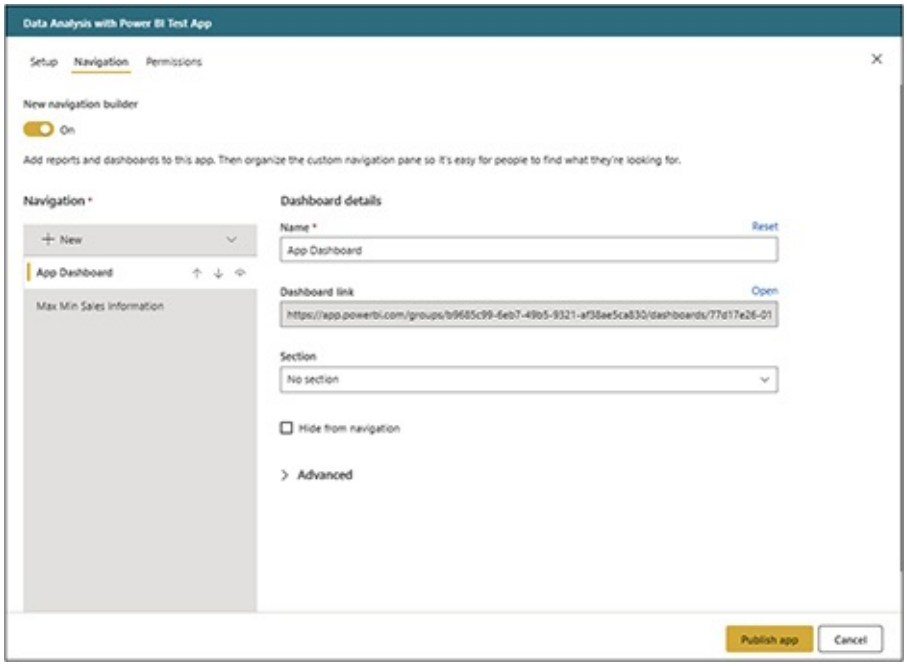

Figure 15-34 *The Navigation page*

27. Select Permissions at the top of the page.
28. Select “Entire organization” to allow the entire organization access to this app. The Permissions page appears as shown in Figure 15-35.

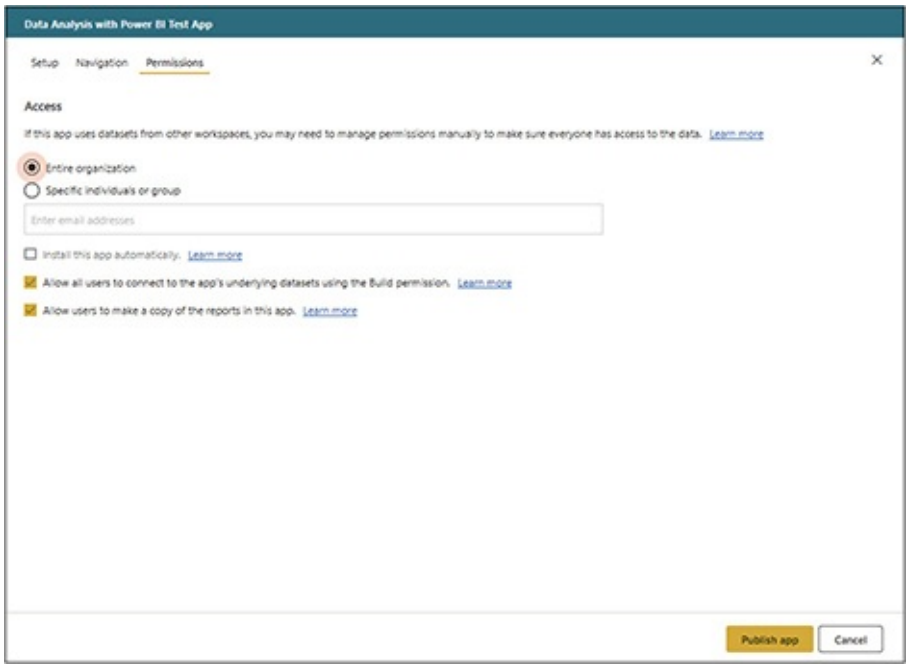

Figure 15-35 *The Permissions page*

29. Click the “Publish app” button. A confirmation dialog box appears.
30. Click Publish.
31. When the “Successfully published” dialog box appears, click Close.

Installing an App

As we did with the content pack, we will play the role of another user who wants to get the dashboard and reports in our app. Our fictional user will find the app we just created and install its content.

1. Click Get Data in the lower-left corner of the page.
2. Click the “My organization” box under “Discover content,” as shown in Figure 15-36.

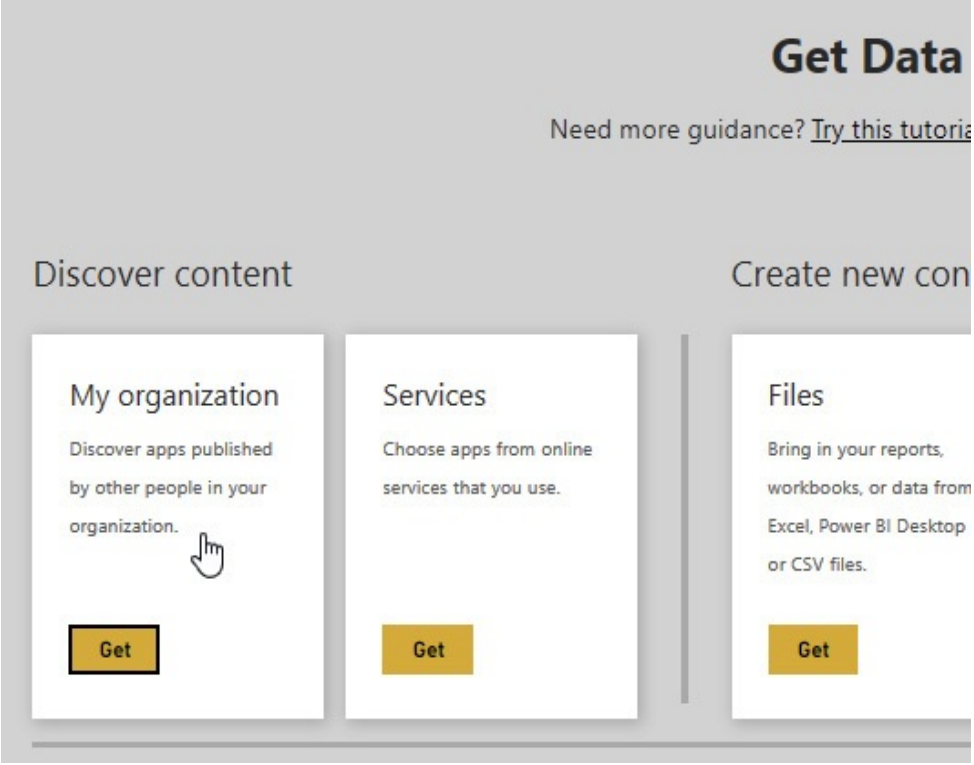

Figure 15-36 *Discover content from My organization*

3. Click the "Get it now" button for "Data Analysis with Power BI Test App," as shown in Figure 15-37.

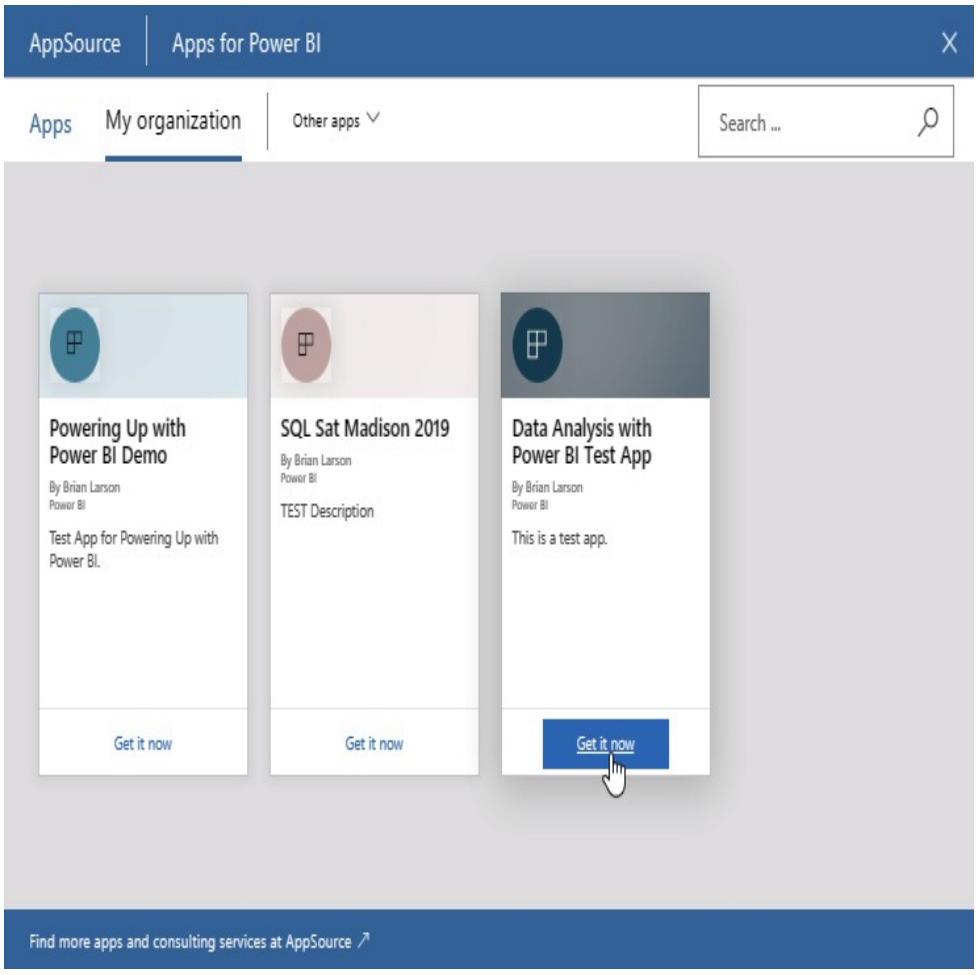

Figure 15-37 *Selecting an app to install*

4. Click the entry for "Data Analysis with Power BI Test App" in the Apps area, as shown in Figure 15-38. You are taken to the dashboard deployed with the app.

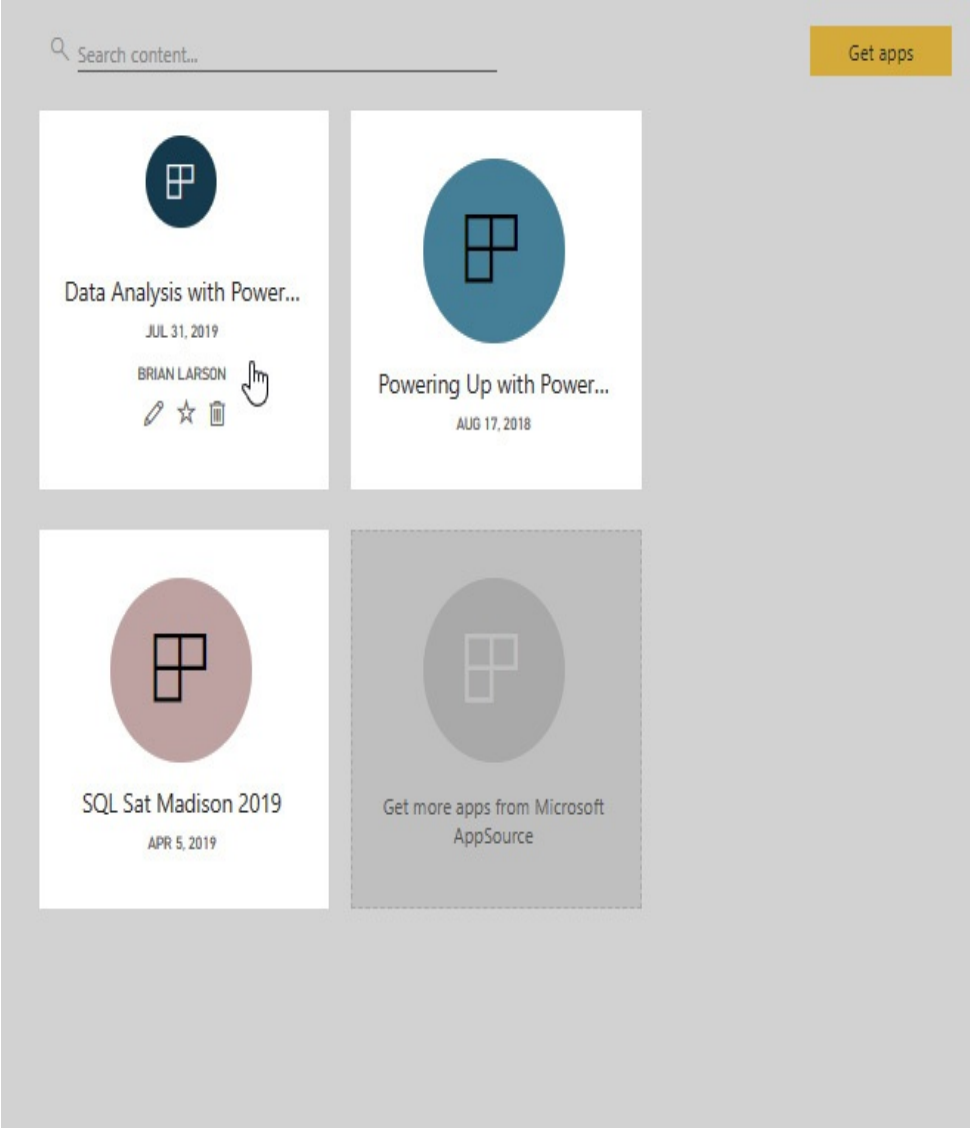

Figure 15-38 *Opening an app*

5. The app content is kept separate from the workspaces. Use the menu on the left to navigate through the app content, as shown in Figure 15-39.

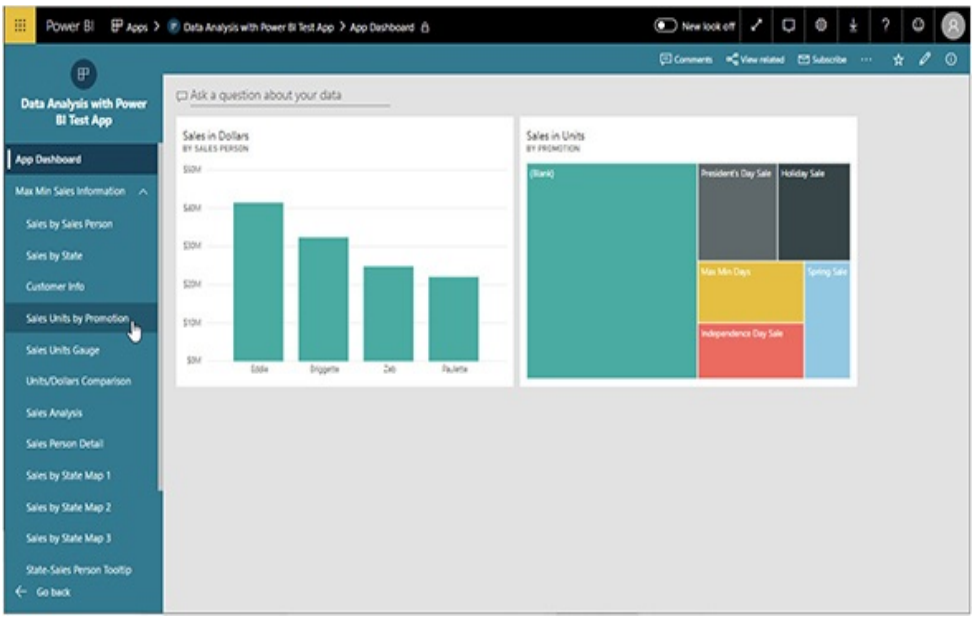

Figure 15-39 *Navigating the app content*

6. Click “Go back” on the left side of the screen to leave the app.
7. On the left side of the screen, click Apps. Note the “Data Analysis with Power BI Test App” is now available on the Apps page.

Deleting an App

1. On the left side of the page, click Workspaces and then click the More (…) button to the right of the “Data Analysis with Power BI Test App” workspace item.
2. Click “Workspace settings,” as shown in Figure 15-40. The Workspace Settings dialog box appears.

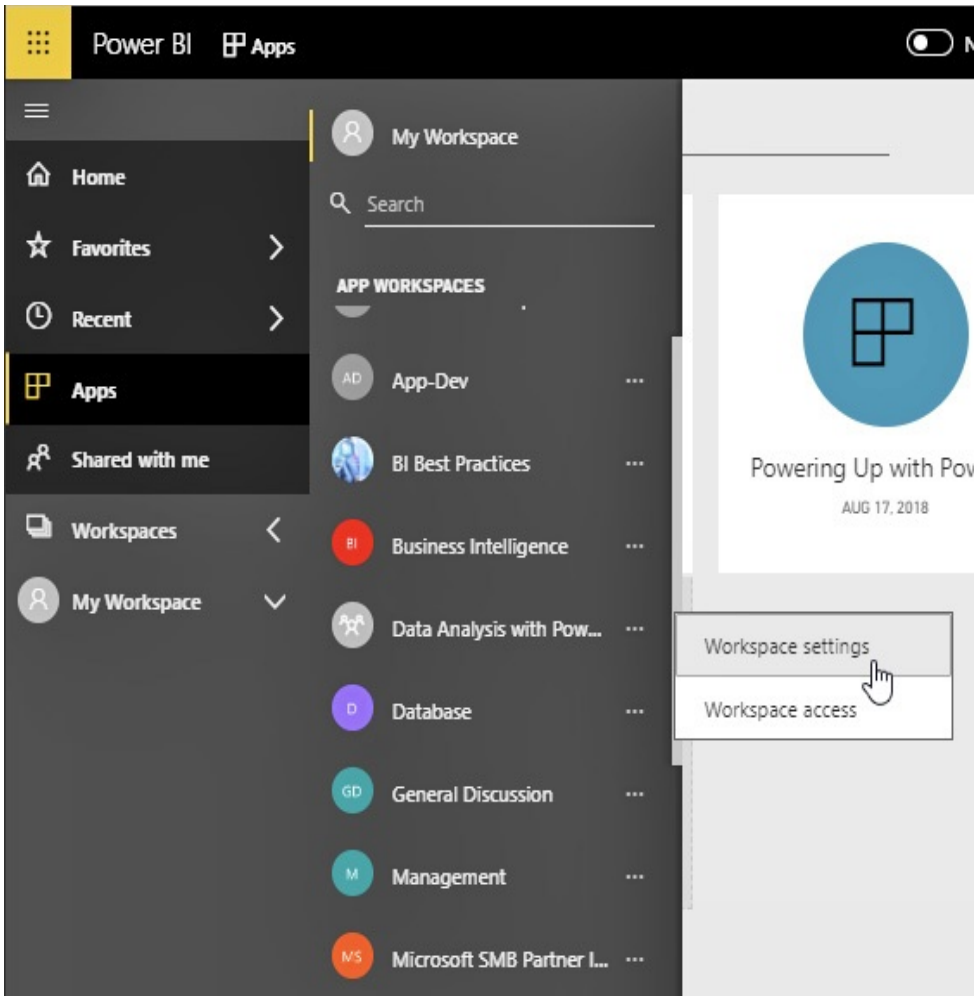

Figure 15-40 *Accessing workspace settings*

3. Click “Delete workspace,” as shown in Figure 15-41. The “Delete workspace?” confirmation dialog box appears.

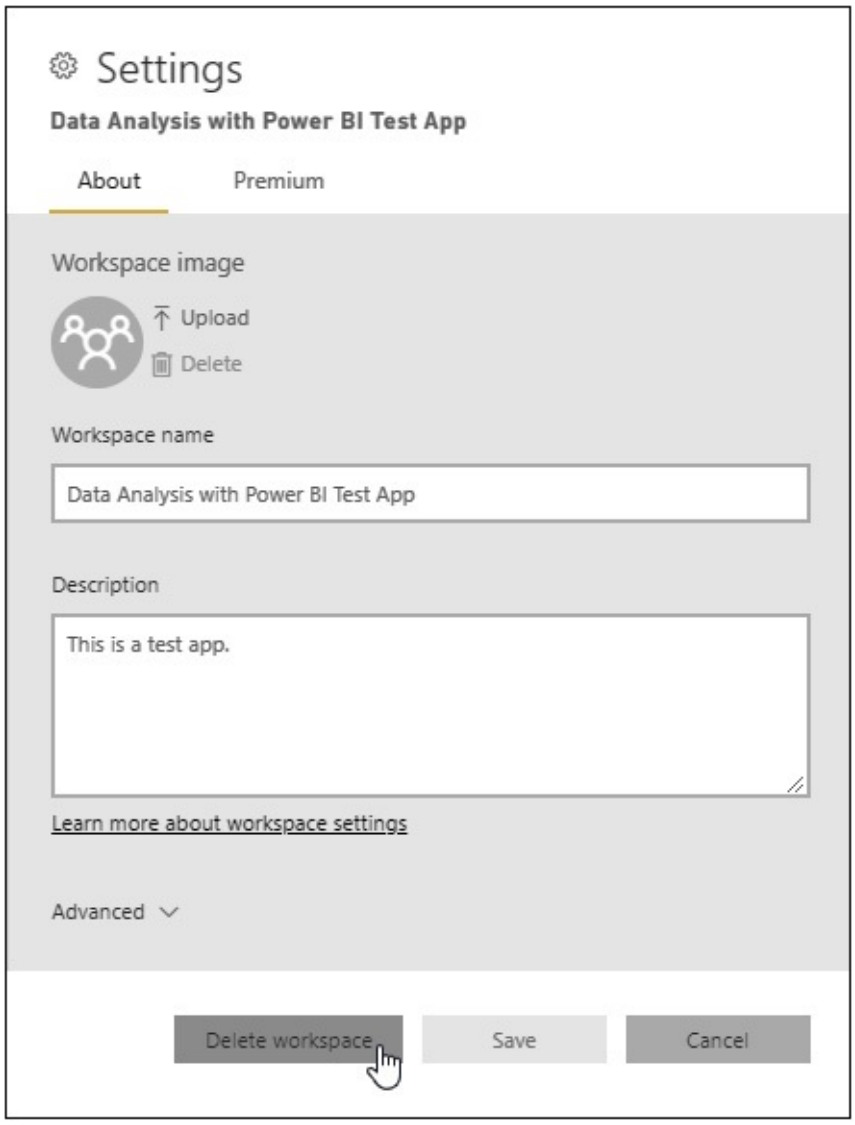

Figure 15-41 *The Workspace Settings dialog box*

4. Click Delete. The workspace and the corresponding app are deleted. Eventually, the app will disappear from our installed apps as well.

THE POWER BI SERVICE AND ROW-LEVEL SECURITY

The data model for our Max Min Sales Information report

includes a custom security role. In order to use this security role in the Power BI Service, we must assign users to this security role. If a user is given access to a report (or its dataset) and is not assigned to a custom security role, they will not have proper security for that report (or that dataset).

Assigning Users to a Custom Security Role

Users are assigned to a custom security role at the dataset level. Once assigned, that user is restricted by that security role. That is true whether they are using a report dependent on that dataset or exploring the data in the dataset directly.

Let's look at how we manage the members of a security role.

Managing Users in a Custom Security Role

1. On the left side of the page, select "My workspace."
2. Select the Datasets tab.
3. In the entry for the Max Min Sales Information dataset, click the ellipsis (…) button.
4. Select Security from the menu, as shown in Figure 15-42. The Row-Level Security page appears.

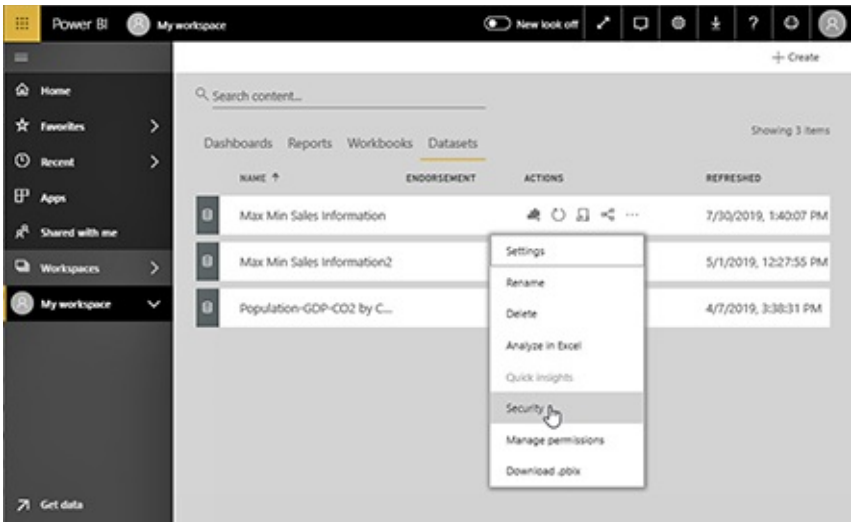

Figure 15-42 *The Security menu option*

5. The “Sales Territory 8” role is selected. Enter the email address of a valid Power BI user and press enter.
6. Click Add. This user is added as a member of the Sales Region 8 role, as shown in Figure 15-43.

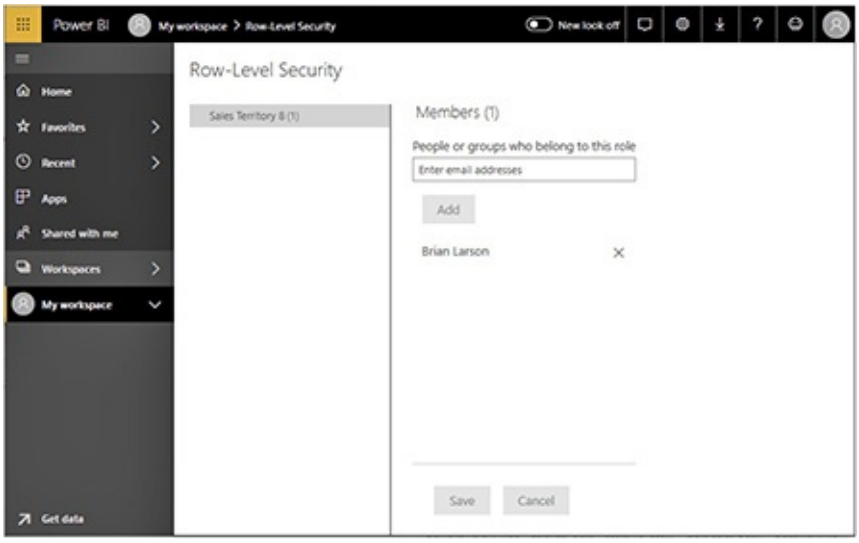

Figure 15-43 *The Sales Territory 8 role with a member*

7. Click Save.
8. To remove this user from the role membership, click the

"X" to the right of the user name.

9. Click Save.

ANOTHER WAY TO SHARE

As an alternative to sharing content in the Power BI Service, we can also share our Power BI reports in an on-premises environment. This is done using the Power BI Report Server. The techniques for sharing and managing Power BI content using the Power BI Report Server are covered in Chapter 16.

Chapter 16

Saving to the Power BI Report Server

In This Chapter

- **Power BI Service and Power BI Report Server Comparison**
- **Installing Power BI Report Server**
- **The Report Catalog**
- **Security**
- **Branding the Power BI Report Server**
- **Powered Up and Ready to Go**

Chapter 15 was all about sharing Power BI content using the Power BI Service (PowerBI.com). The Power BI Service contains a number of features to enhance the Power BI user experience. We can combine visualizations from multiple reports to create dashboards. We can easily access data stored on Microsoft Azure platforms. We can even leverage the On-premises Data Gateway to securely access on-premises data.

There is one thing, however, we can't get away from when using the Power BI Service, and that is the fact that it exists on

cloud-based infrastructure. There are some organizations that are not ready, for one reason or another, to build their reporting platform in the cloud. Others don't mind having part of their reporting infrastructure in the cloud, but need to complement the cloud capabilities with an on-premises infrastructure. For those organizations, there is the Power BI Report Server.

POWER BI SERVICE AND POWER BI REPORT SERVER COMPARISON

As noted in Chapter 3, the Power BI Report Server is a customized version of SQL Server Report Server. This is the server for delivering SQL Server Reporting Services (SSRS) reports throughout an enterprise. The Power BI Report Server retains all the features of SQL Server Reporting Services and adds to it the ability to process data in Power BI tabular models and render Power BI reports.

The Power BI Report Server is licensed through Power BI Premium or through a SQL Server Enterprise Edition license as part of a Microsoft Enterprise Agreement/Software Assurance (EA/SA). Be sure to check on licensing before pursuing a Power BI Report Server installation. In some cases, it can be a very expensive alternative to the Power BI Service.

Versions

Microsoft upgrades the Power BI Report Server frequently—about every three to five months. That is an aggressive upgrade cadence for an on-premises server application. Still, it's almost a snail's pace when compared to the monthly

updates to the Power BI Service. Therefore, the Power BI Report Server is often several months behind the Power BI Service when it comes to feature updates. You may have to wait a bit for the latest features supported by the Power BI Service to reach a new version of the Power BI Report Server.

Power BI Desktop Optimizations

As noted in Chapter 3, there are two optimizations of Power BI Desktop—one optimized for Power BI Service and the other optimized for the Power BI Report Server. Because the Power BI Report Server is often several months behind the Power BI Service in feature support, that means the Power BI Desktop optimized for the Power BI Report Server is behind the Power BI Desktop optimized for the Power BI Service. As of the writing of this chapter, the Power BI Service and associated Power BI Desktop were on the July 2019 release. The Power BI Report Server and its associated Power BI Desktop were on the May 2019 release.

Not surprisingly, the best way to obtain the appropriate version of Power BI Desktop optimized for the Power BI Service is through the Power BI Service. Likewise, the best way to obtain the appropriate version of Power BI Desktop optimized for Power BI Report Server is through the link on your local Power BI Report Server. This is shown in Figure 16-1.

Figure 16-1 *Link for downloading the Power BI Desktop Report Server optimizations*

INSTALLING POWER BI REPORT SERVER

Installing an instance of Power BI Report Server is a two-step process. The Power BI Report Server is installed by running an executable downloaded from PowerBI.com. Once the installation is done, the Report Server Configuration Manager is used to complete the final setup of the new Power BI Report Server instance.

The Power BI Report Server Installation Executable

The Power BI Report Server install program can be found using the Products menu at the top of the PowerBI.com landing page (prior to signing in). Selecting "Power BI Report Server" from the Products menus takes you to the Power BI Report Server information page. I find it works best to click the "Advanced download options" link and then download the PowerBIReportServer.exe installation executable from there.

Download and run (open) the executable file. Follow the steps to complete the installation. When the installation is done, you can use the button on the last screen of the installation program to launch the Report Server Configuration Manager.

The Report Server Configuration Manager

In addition to launching the Report Server Configuration Manager at the end of the installation process, you can also launch the Report Server Configuration Manager from the Start Menu. You will find it on the Start Menu in the Microsoft Power BI Report Server folder. This provides a way to return to the Report Server Configuration Manager if you need to make configuration changes down the road.

When the Report Server Configuration Manager is launched, it asks for the name of a server to connect to. Once you enter a server name and click Find, the program finds all instances of Power BI Report Server (or SQL Server Reporting Services) running on that server and displays them in the Report Server Instance dropdown box. You need to select an instance and click Connect to enter the utility program with the configuration information for that Report Server instance loaded. (These will be filled in by default during your initial installation/configuration process.)

The Report Server Configuration Manager contains a number of pages, each geared toward configuring a different aspect of the Report Server.

Report Server Status Page

The Report Server Status page displays status information about the Report Server instance you selected. This page provides buttons to start or stop this Report Server instance.

Service Account Page

The Service Account page enables you to view and change the credentials used to run the Report Server service.

Web Service URL Page

The Web Service URL page enables you to create or change the URL used by the Report Server web service. Power BI Desktop uses the Report Server web service to communicate with the Report Server when deploying reports. Use this page to configure the ports to be used when communicating with the Report Server web service and to specify whether a secure connection is required.

Report Server Database Page

The Report Server Database page enables you to select the set of databases that will serve as the Report Catalog. The Report Catalog stores all the configuration information and the reports saved to this Report Server. You can select the database server name, the name of the database on that server, and the credentials used to connect to that server. A SQL Server instance is required to host the Report Catalog.

Web Portal URL Page

The Web Portal URL page enables you to view and change the name of the virtual directory used by the web portal. The web portal serves as the main user interface for interacting with the Report Server.

E-mail Settings Page

The E-mail Settings page enables you to identify an SMTP server that can be used by the Report Server. The SMTP server is used for delivering report subscriptions via e-mail. At the time of this writing, subscriptions could be used with SQL Server Reporting Services reports on the Report Server but were not available for Power BI reports.

Execution Account Page

The Execution Account page enables you to specify a set of login credentials to be used by the Report Server when it needs to access a file or other resource.

Encryption Keys Page

The Report Server uses an encryption key to encrypt and store sensitive information, such as credentials. You can use the Backup button to create a backup copy of the encryption key. This guards against corruption of the encryption key, which would cause all of the sensitive information stored on the Report Server to becoming unreadable.

Subscription Settings Page

The Subscription Settings page is used to specify credentials to be used by subscriptions to allow it to gain rights to create a file in a file share. At the time of this writing, subscriptions could be used with SQL Server Reporting Services reports on the Report Server but were not available for Power BI reports.

Scale-out Deployment Page

The Scale-out Deployment page is used to add servers to a scale-out installation of the Power BI Report Server. A scale-out deployment allows us to spread the rendering load across multiple servers. Each server added to the scale-out list uses the same encryption key. In this way, encrypted data stored in the common Report Catalog can be decrypted by any Report Server in the scale-out installation.

Power BI Integration Page

SQL Server Reporting Services reports can be pinned to a Power BI dashboard hosted on PowerBI.com. In order for PowerBI.com to display these reports for a user, it must know

how to find a Report Server that can render the reports. The Power BI Integration page is used to register this Report Server with PowerBI.com. To register, simply enter your PowerBI.com credentials.

NOTE

This feature has nothing to do with rendering Power BI reports. It is used solely to support the ability of PowerBI.com to host SQL Server Reporting Services reports.

Menu Bar

The Connect button, at the top of the page's menu area, lets you connect to a different server and then select a Report Server instance on that server.

THE REPORT CATALOG

Before we deploy reports to the Power BI Report Server, we need to have an understanding of the way the Report Server organizes reports. In the Power BI Service, our content was organized in workspaces and apps. On the Power BI Report Server, content is stored in the Report Catalog and organized in folders.

Folders

In the Report Catalog, reports are arranged into a system of folders similar to the Windows or Mac file system. Folders can contain reports, supporting files, and even other folders. The easiest way to create, view, and maintain these folders is

through the web portal.

Although the Report Catalog folders look and act like Windows file system folders, they are not actual file system folders. You cannot find them anywhere in the file system on the computer running the Report Server. Report Catalog folders are screen representations of records in the Report Catalog database.

Each folder is assigned a name. Folder names can include just about any character, including spaces. However, folder names cannot include any of the following characters:

; ? : @ & = + $, \ * < > | " /

Also, a folder name cannot consist exclusively of dots or spaces.

In addition to a name, folders can be assigned a description. The description can contain a long explanation of the contents of the folder. It also can help users determine what types of reports are in a folder without having to open that folder and look at the contents. Both the folder name and the description can be searched by a user to help them find a report.

The Web Portal

The web portal provides a straightforward method for creating and navigating folders in the Report Catalog. When you initially install the Power BI Report Server, the Home folder is created by default. This is the only folder that exists at first.

The default Uniform Resource Locator (URL) for accessing the web portal site on the computer running the Power BI Report Server is

```
http://ComputerName/reports
```

where *ComputerName* is the name of the computer on which the Power BI Report Server was installed. If you are using a secure connection to access the web portal, replace **http:** with **https:**. If you are on the same computer where the Power BI Report Server is running, you can use the following URL:

```
http://localhost/reports
```

No matter how you get there, when you initially access the web portal, it appears similar to Figure 16-2.

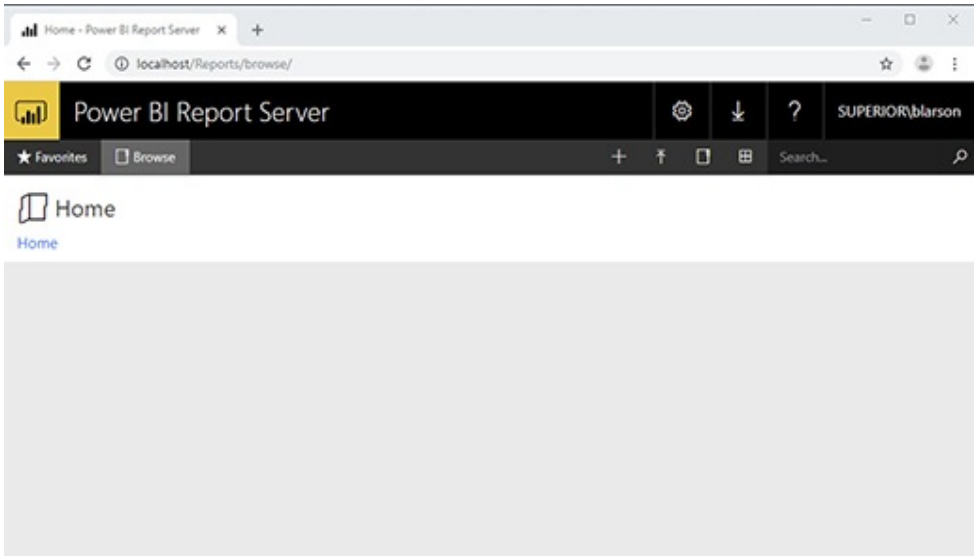

Figure 16-2 *The web portal with no folders defined*

Notice the URL shown in Figure 16-2 is a bit different from the URLs given previously. This is because the web portal web application redirects you to browse mode. The browse mode is used to display folder contents.

NOTE

Figure 16-2 shows the web portal as it appears for a user with content manager privileges. If you do not see the same items in the toolbar, you do not have content manager privileges.

The web portal follows the HTML5 standard. This should get you great compatibility across all modern browsers. So feel free to use any modern browser when working with the web portal. To use the web portal, you must have scripting enabled in your browser.

Adding a New Folder Using the Web Portal

To create a new folder, click the New button in the web portal toolbar and select Folder from the menu, as shown in Figure 16-3. The “Create a new folder” dialog box will appear. Enter the name of the folder you wish to create. The dialog box will appear similar to the one shown in Figure 16-4.

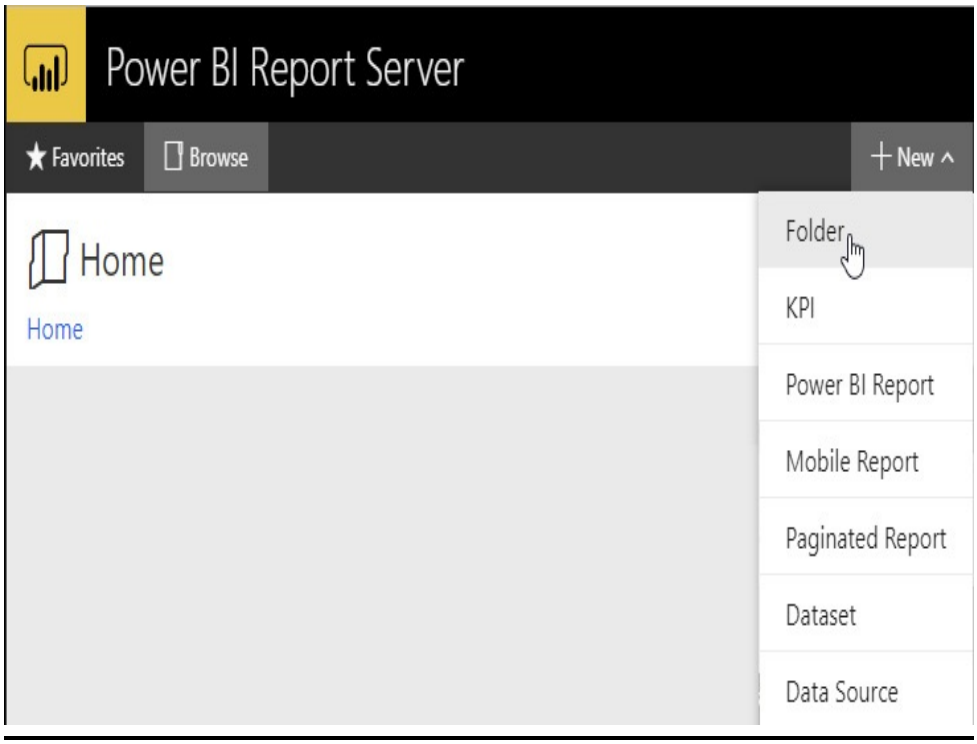

Figure 16-3 *The Folder menu option*

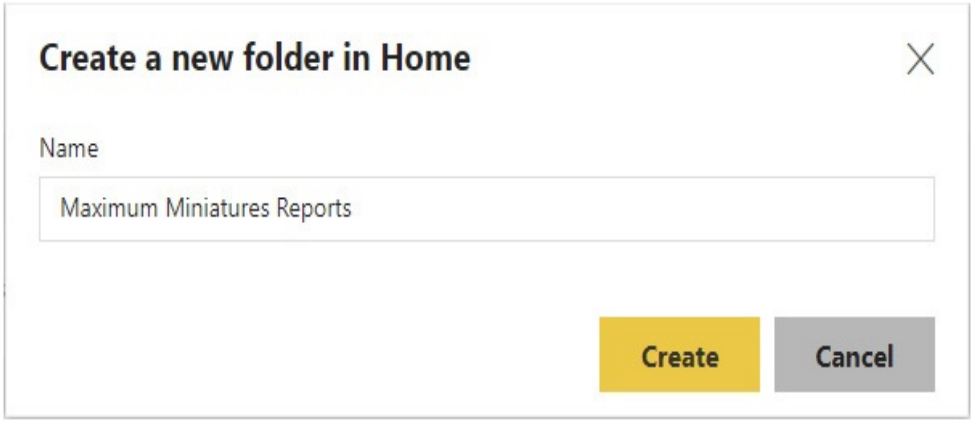

Figure 16-4 *The "Create a new folder" dialog box*

Folder Properties

We noted earlier that each folder can also have a description. To add a description to a folder, click the ellipsis (…) button for the folder, as shown in Figure 16-5. Select Manage from the popup box. This will display the Properties page for the

folder. You can enter the folder description.

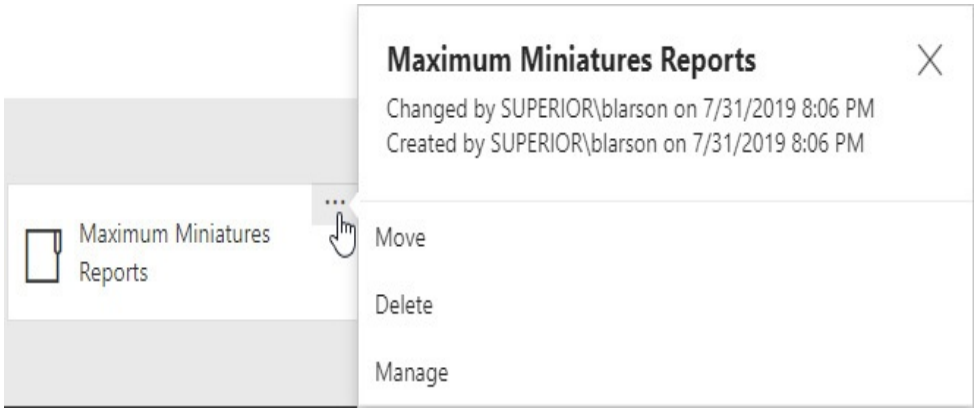

Figure 16-5 *The Folder menu*

There is also a check box titled “Hide this item.” When this box is checked, the folder is not visible when the folder contents are displayed in tile view. This hidden option plays a role with the SQL Server Reporting Services content that the web portal user interface was originally designed to work with. It does not really come into play when working with Power BI content.

The folder Properties page is shown in Figure 16-6.

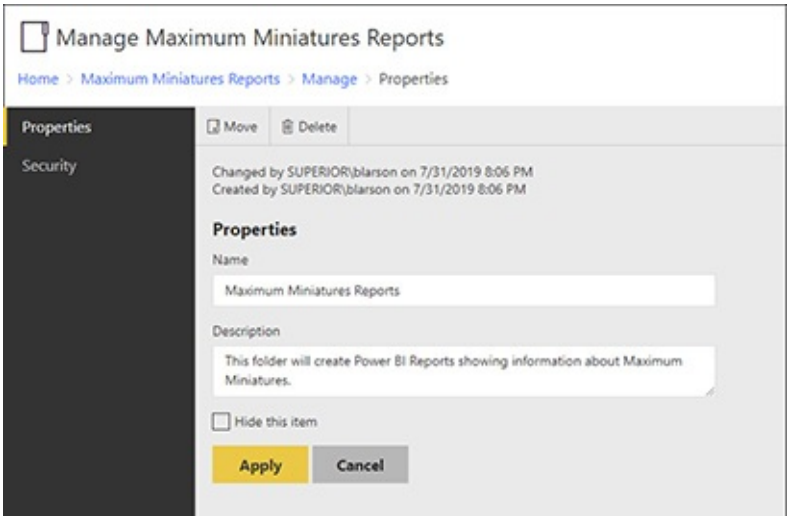

Figure 16-6 *The folder Properties page*

Navigating Folders

To view the contents of the new folder, click the folder name. The name of the current folder appears in bold text near the top of the page. Below the name of the current folder is the path from the Home folder to the current folder. This is the breadcrumb path you can use to return to any folder you clicked through to get to the current location. To return to any folder in the current path, click that folder name in the breadcrumb path. You can return to the Home folder by clicking Home at the beginning of the breadcrumb path, as shown in Figure 16-7. You can also return to the Home folder by clicking the main heading or logo in the upper-left corner of the page. By default, the main heading is “Power BI Report Server.” Later in this chapter, you will see how to customize this heading to put your own brand on the web portal.

Figure 16-7 *Navigating to the Home folder using the breadcrumbs*

Saving a Report to the Power BI Report Server

Once we have created an appropriate folder structure to hold our Power BI reports, we can save our reports to the Power BI

Report Server. This is done using the version of Power BI Desktop optimized for the Power BI Report Server. We simply save the report file somewhere in the Power BI Report Server folder structure.

With a Power BI .pbix file open in the Power BI Desktop version optimized for the Power BI Report Server, select File | Save as | Power BI Report Server, as shown in Figure 16-8. The Power BI Report Server Selection dialog box appears. This is shown in Figure 16-9.

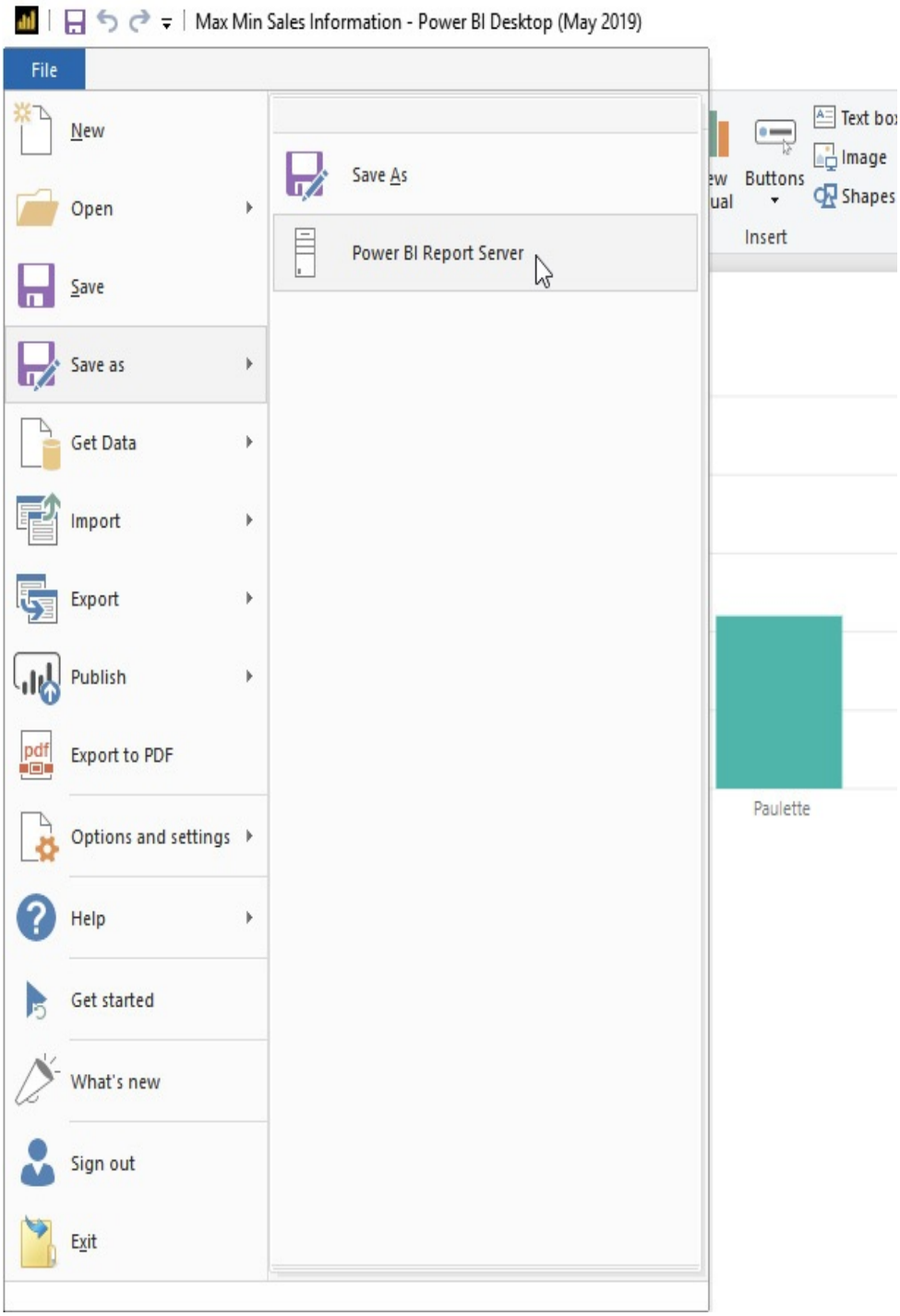

Figure 16-8 *Saving a Power BI report to the Power BI Report Server*

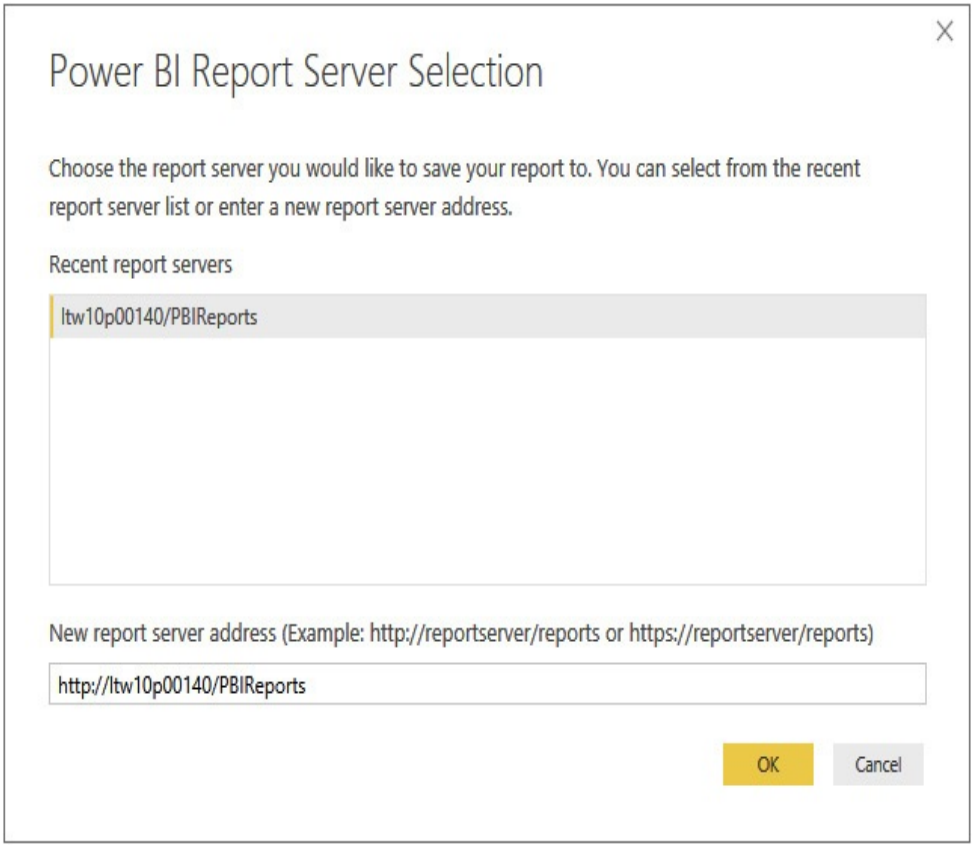

Figure 16-9 *The Power BI Report Server Selection dialog box*

The Power BI Report Server Selection dialog box shows the URLs for any Power BI Report Servers we have saved reports to in the past. The bottom portion of this dialog box has a text entry area where we can enter the URL for a new server. When the desired server is selected, we can click OK to continue.

The "Save report" dialog appears next, as shown in Figure 16-10. The Save report dialog box starts out in the Home folder of the selected server. It shows the folders and the Power BI reports in the folder being displayed. We can double-click a folder name to navigate into that folder. We can click the up arrow next to the folder path to navigate back up a

level if we went into the wrong folder. Once in the correct folder, we click OK to save the report.

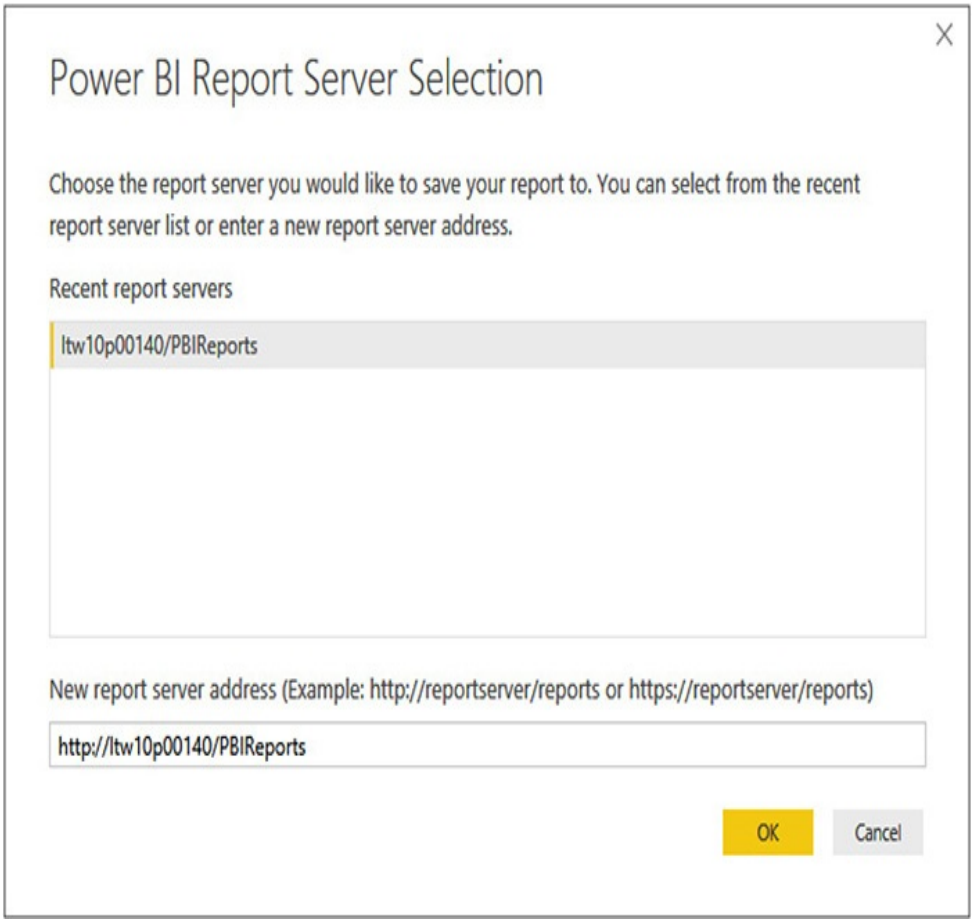

Figure 16-10 *The "Save report" dialog box*

Because this is a "save as" operation, the folder on the Power BI Report Server becomes the last saved location for this report. If we make changes to the report and click File | Save or click the Save button at the top of the window, the changes are saved to the Power BI Report Server, not to the original location in the file system.

NOTE

The Open menu in the version of Power BI Desktop optimized for Power BI Report Server supports opening a Power BI report from the Report Server.

The Power BI Report Server caches its pages for efficiency. That means your newly saved reports may not show up in the folder on the Power BI Report Server. You may need to use the browser's refresh button to get the Power BI Report Server to update the content of the folder being displayed in order to see a newly saved report.

Once the report is visible in the folder, you click the report and interact with it just as you would in the Power BI Service. This is shown in Figure 16-11.

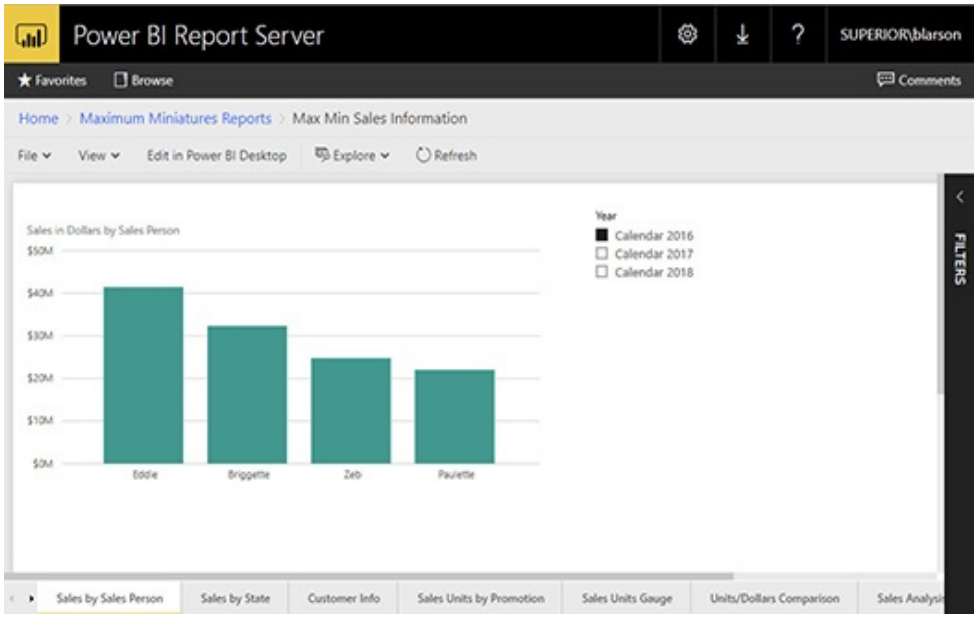

Figure 16-11 *Viewing a report on the Power BI Report Server*

NOTE

For Power BI reports that include roles, you may need to configure row-level security on the Power BI Report Server before the content of the report will display. See the “Row-Level Security” section of this chapter for more information.

SECURITY

Security may be set at two levels on a Power BI Report Server. First, security is set on folders and reports to determine who can explore the folders and execute the reports as well as who can manage these items. Second, row-level security can be configured to use security roles created within a Power BI data model.

Folder and Report Security

In the Power BI Report Server, security was designed with both flexibility and ease of management in mind. Flexibility is provided by the fact that individual access rights can be assigned to each folder and to each report within a folder. You can specify exactly who has rights to each item and exactly what those rights are. Ease of management is provided by security inheritance, security roles, and integration with Windows security. We begin our discussion with the last entry in this list.

Integration with Windows Security

The Power BI Report Server does not maintain its own list of

users and passwords. Instead, in its default configuration, it depends entirely on integration with Windows security. When a user accesses the Power BI Report Server, that user must authenticate with the Report Server. In other words, the user must have either a valid domain user name and password or a local user name and password to log on to the Report Server.

NOTE

If it is impossible for each report user to have their own credentials on the Power BI Report Server, it is possible to configure the server to use forms-based security through a custom security extension.

Once this logon occurs, the Power BI Report Server utilizes the user's name and group memberships to determine what rights the user possesses. The user can access only those folders and reports to which they have rights. In the web portal, users do not even see the folders they cannot browse and reports they cannot run. There is no temptation for the user to try and figure out how to get into places they are not supposed to go, because they do not even know these places exist.

Local Administrator Privileges

In most cases, rights must be explicitly assigned to folders and items. There is, however, one security assignment that is created by default. Any user who is a member of the local administrators group on the computer hosting the Power BI Report Server has content manager rights to all folders and all

items. This is done through the BUILTIN\Administrators designation on the Report Server.

Viewing the Security Rights for a Folder

To view the rights for a given folder, we look at the properties of that folder. Earlier, you learned we could do this by clicking the ellipsis (…) button for that folder and selecting Manage from the popup menu. We can view the properties of the folder whose contents we are currently viewing by clicking the Folder Properties button, as shown in Figure 16-12.

Figure 16-12 *The Folder Properties button*

Either method takes us to the Folder Properties. Click the Security tab on the left side of the page to see the security settings. This is shown in Figure 16-13.

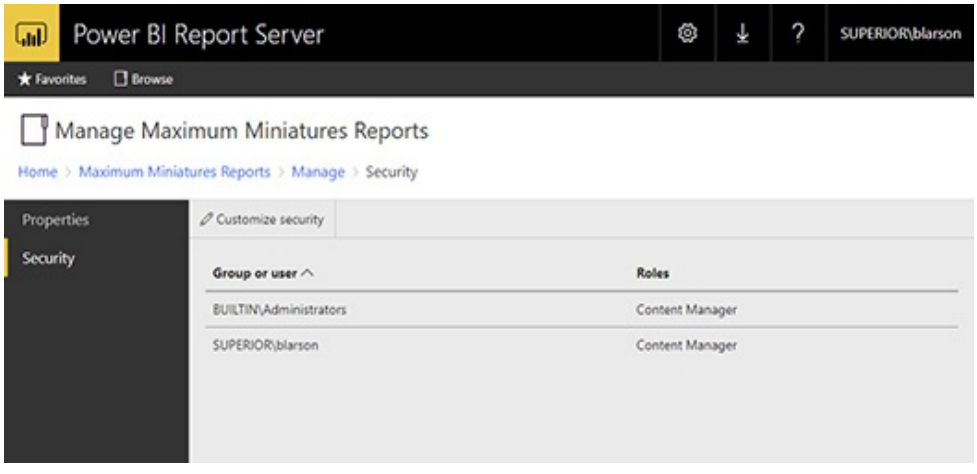

Figure 16-13 *The Security tab of the Properties page*

Security Roles

The rights to perform tasks are grouped together to create *roles*. Power BI Report Server includes a number of predefined roles to help you with security management.

The Browser Role The Browser role is the basic role assigned to users who are going to view reports but who are not going to create folders or upload new reports. The Browser role has rights to perform the following tasks:

- View folders
- View reports

The Publisher Role The Publisher role is assigned to users who are going to create folders and upload reports. The Publisher role does not have rights to change security settings. The Publisher role has rights to perform the following tasks:

- Manage folders
- Manage reports

The My Reports Role The My Reports role is designed to be used only with a special folder called the My Reports folder. Within this folder, the My Reports role gives the user rights to do everything except change security settings. The My Reports role has rights to perform the following tasks:

- Manage folders
- Manage reports
- View folders
- View reports

The Content Manager Role The Content Manager role is assigned to users who are managing the folders and reports. The Content Manager role has rights to perform all tasks, excluding system-wide tasks.

The Report Builder Role The Report Builder role is a holdover from SQL Server Reporting Services.

The System User Role The system-wide security tasks have two predefined roles. The System User role has rights to perform the following system-wide tasks:

- Execute reports
- View report server properties

The System Administrator Role The System Administrator role provides the user with rights to complete any of the tasks necessary to manage the Power BI Report Server. This role has rights to perform the following system-wide tasks:

- Execute reports
- Manage report server properties
- Manage report server security
- Manage roles

Creating Security Role Assignments

As stated previously, role assignments are created when a Windows user or a Windows group is assigned a role for a folder or a report. Role assignments are created on the Security tab of the Properties page for the folder or report. These role assignments control what the user can see within a folder and what tasks the user can perform on the folder or report.

If the Security tab of the Properties page includes the "Customize security" button, as shown earlier in Figure 16-13, then the security settings for that folder or report are being inherited from its parent folder. Clicking the "Customize security" button will break that inheritance and enable you to make security role assignments for this item. When you click the "Customize security" button, you will see the confirmation dialog box shown in Figure 16-14. Click OK to confirm.

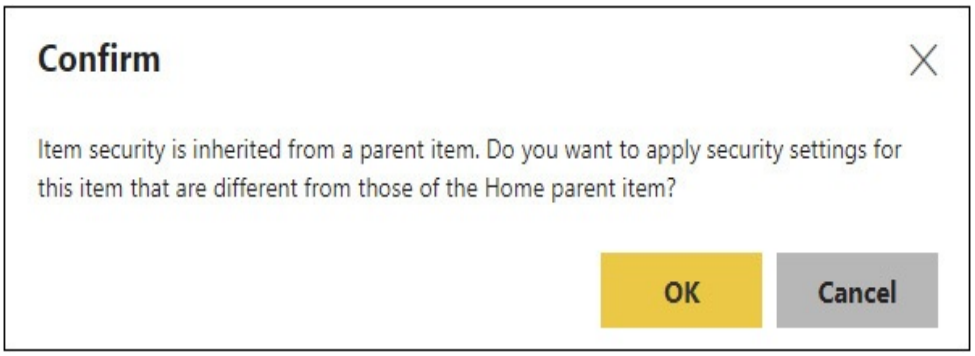

Figure 16-14 *The Confirm dialog box for breaking security inheritance*

When a folder or report is not inheriting security settings from its parent folder, the Security tab of the Properties page appears as shown in Figure 16-15. We can click the "Add group or user" button to create a new security assignment. When we do, we see the New Role Assignment page.

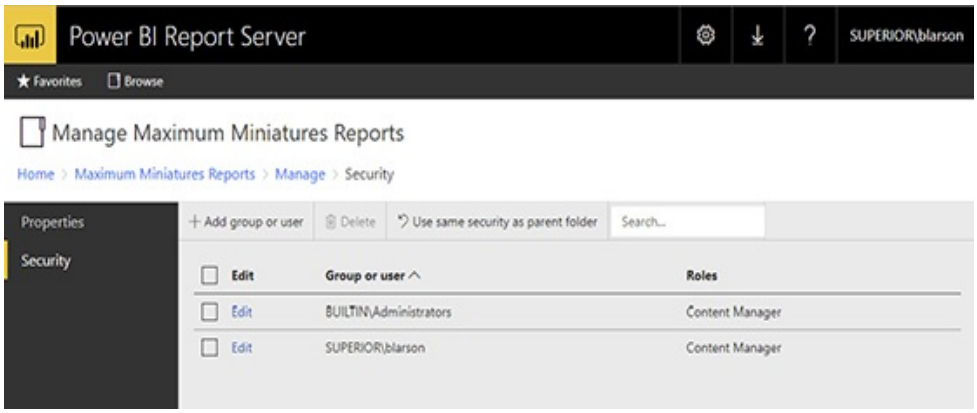

Figure 16-15 *The Security tab when security settings are not inherited*

On the New Role Assignment page, the group or user must be a valid domain user or group, or a valid local user or group. When using a domain user or group, this must be in the format *DomainName\UserName* or *DomainName\GroupName*. If you are using a local user or group, you can simply type the user name or group name. Once the user or 500group is specified, you can select the role to assign to this user or group. The completed New Role Assignment page should appear similar to Figure 16-16.

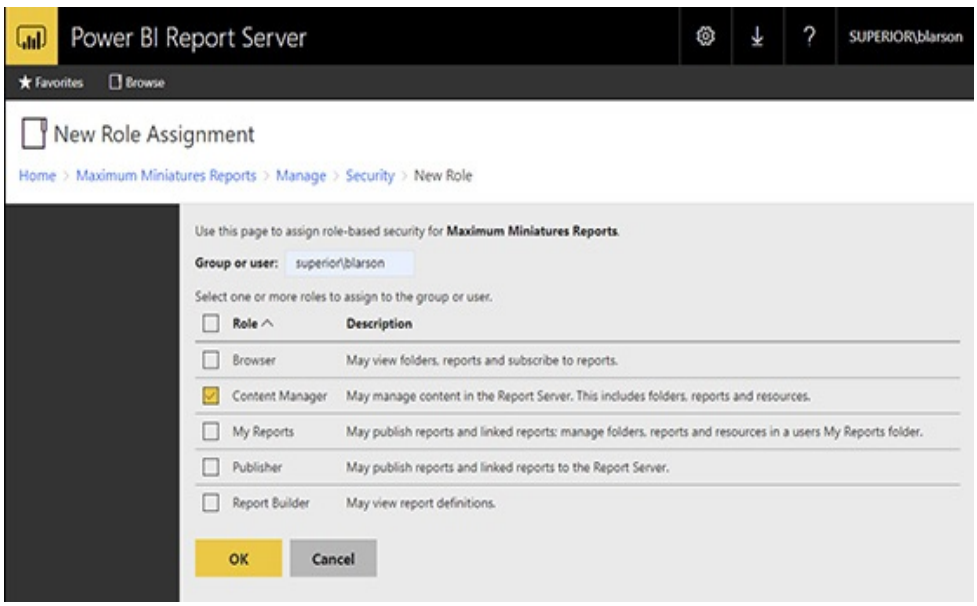

Figure 16-16 *The New Role Assignment page*

Site-Level Security Role Assignments

Site-level security role assignments are created on the "Site settings" page. Click the Settings button (the cog icon) and select "Site settings" from the menu, as shown in Figure 16-17. The "Site settings" page appears. Click Security on the left to view the Security tab of the "Site settings" page, as shown in Figure 16-18. The "Add group or user" button is used to create new security role assignments in the same manner as at the folder or report level.

Figure 16-17 *The "Site settings" menu option*

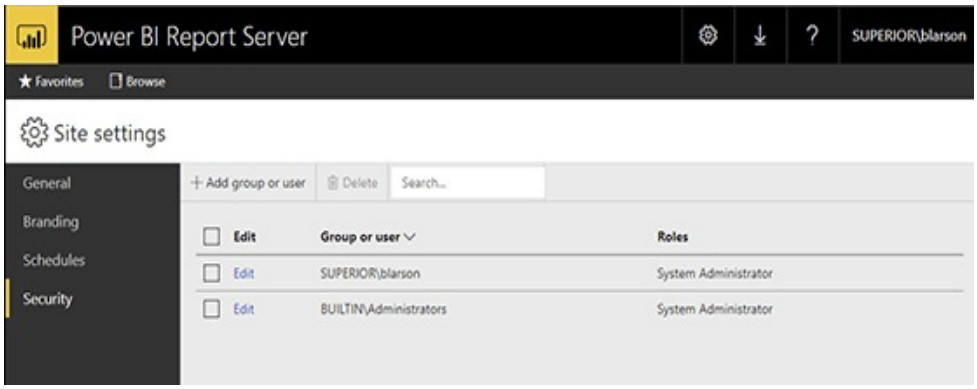

Figure 16-18 *The Security tab of the "Site settings" page*

Role Assignments Using Windows Groups

As mentioned previously, role assignments can be made to domain users or to domain groups. If you create your role assignments using Windows users, you need to create a new

set of role assignments every time a new user needs to access the Power BI Report Server. This can be extremely tedious if you have a complex set of role assignments for various folders, reports, and resources.

In most cases, creating role assignments using domain groups is better. Then, as new users come along, you simply need to add them to the domain group that has the appropriate rights in Power BI Report Server. This is much easier!

Row-Level Security

We saw in Chapter 15 that we had to assign users to our custom row-level security role in the Power BI Service environment. The same is true for the Power BI Report Server. We must assign users to the custom security role in order for that role to limit their data access.

On the Power BI Report Server, the role assignments are done at the report level. To add row-level security to a report, click the ellipsis (…) button for the report, as shown in Figure 16-19, and select Manage from the popup box. This will display the Properties page for the report.

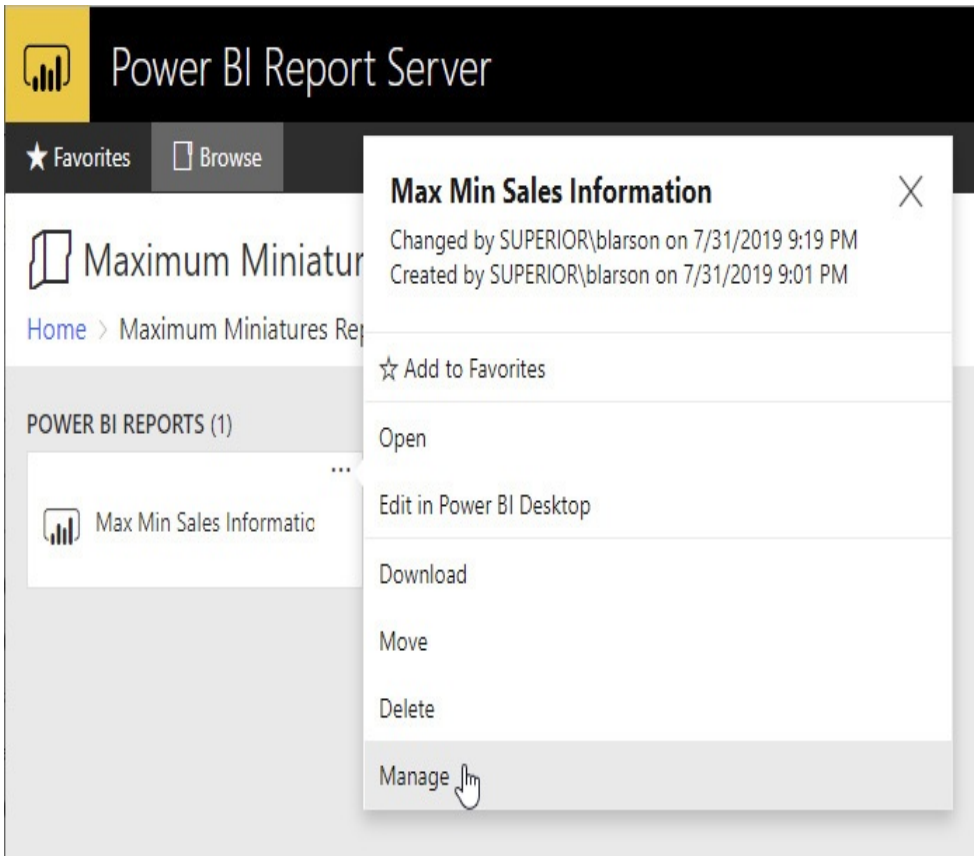

Figure 16-19 *The report menu*

On the left side of the page, select “Row-level security” to display the row-level security page. Click Add Member to add a member to a security role, as shown in Figure 16-20. The Add Member page appears.

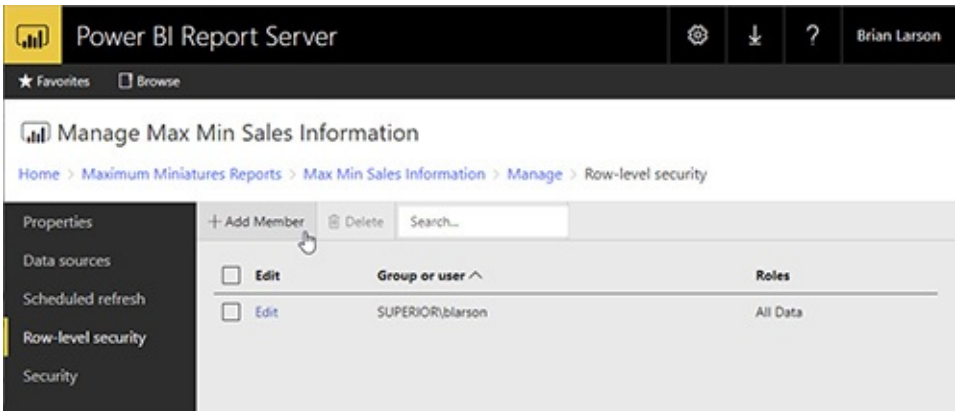

Figure 16-20 *The "Row-level security" page*

In the "Group or user" area, enter the name of a valid domain user or group, or a valid local user or group. Check the box for the row-level security role or roles to assign to this user or group. This is shown in Figure 16-21. Click OK when complete.

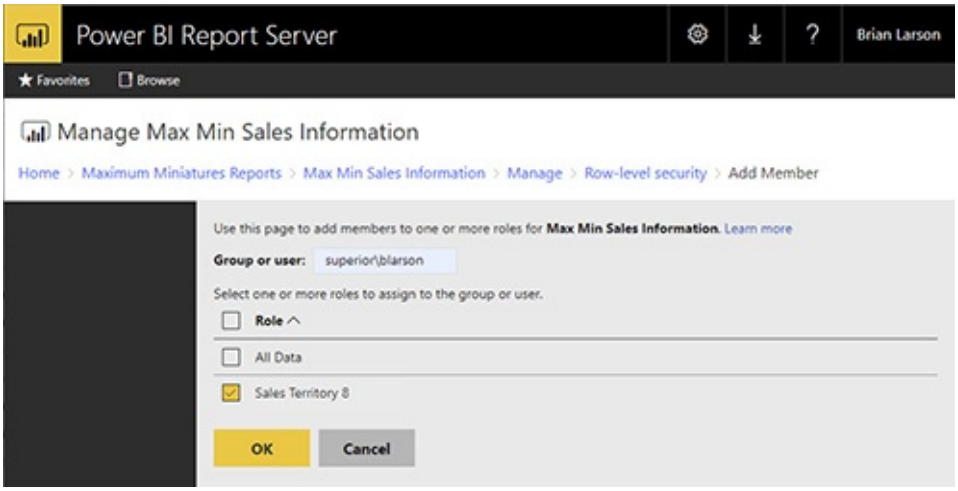

Figure 16-21 *Adding a user to a row-level security role*

BRANDING THE POWER BI REPORT SERVER

Branding enables us to customize the Power BI Report Server to better match our organization. There is one very quick and easy change we can make to provide a small customization and a larger change we can make for more comprehensive branding. We will take a look at both.

Modifying the Site Name

The small customization involves changing the name of the

Power BI Report Server site. This is done on the General tab of the “Site settings” page. Use the “Site settings” menu option to navigate to the “Site settings” page, as shown previously in Figure 16-17. The General tab of the “Site settings” page is shown in Figure 16-22.

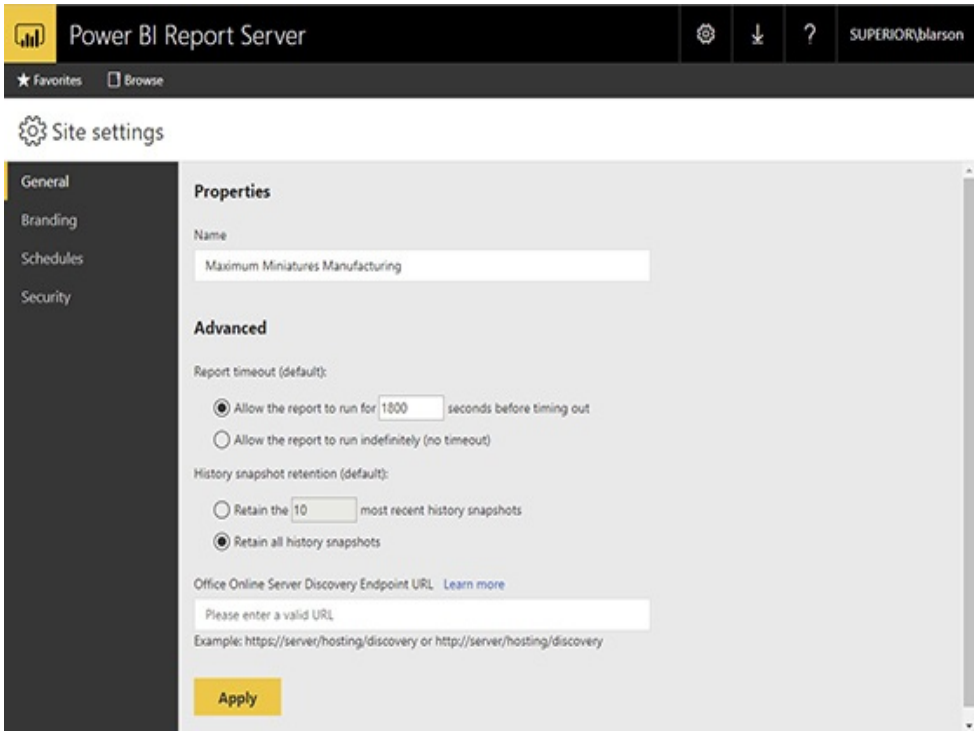

Figure 16-22 *The General tab of the “Site settings” page*

A custom site name is shown in Figure 16-22. Once we click Apply, the custom site name is shown in the upper-left corner of every Power BI Report Server page. This is shown in Figure 16-23.

Figure 16-23 *The Power BI Report Server with a custom site name*

Creating a Brand Package

The Branding tab of the "Site settings" page enables us to load a brand package to customize the site. The brand package consists of three items packaged in a ZIP file:

- **Metadata.xml** A table of contents for the package
- **Colors.json** Custom color definitions for the site
- **Logo file** An image file to use in place of the site name (optional)

Metadata.xml

The Metadata.xml file serves as the table of contents for the brand package. It provides references to the other two files in the package. Here is the format of the Metadata.xml file:

```
<?xml version="1.0" encoding="utf-8"?>
<SystemResourcePackage
   xmlns="http://schemas.microsoft.com/sqlserver/reporting/2016/01/
                                          systemresourcepackagemetadata"
   type="UniversalBrand" version="2.0.2" name="Data Analytics Brand Package">
 <Contents>
   <Item key="colors" path="colors.json" />
   <Item key="logo" path="Branding Logo.jpg" />
 </Contents>
</SystemResourcePackage>
```

(In the actual Metadata.xml file, `systemresourcepackagemetadata` should be on the same line as the rest of the xmlns property value.)

The name property in the Metadata XML structure can be set to whatever you like in order to identify your brand package. The path property associated with the logo file must be the name of the logo file loaded into the brand package. If no logo is included in the brand package, the entire Item element for the logo should be omitted from the Metadata.xml file.

Colors.json

The Colors.json file consists of sets of name/value pairs used to generate the cascading style sheet (CSS) that ultimately controls the look of the web portal. The Colors.json file will look similar to the following:

```
{
    "name":"Default brand",
    "version":"1.0",
    "interface":{
        "primary":"#bb2124",
        "primaryAlt":"#d31115",
        "primaryAlt2":"#671215",
        "primaryAlt3":"#bb2124",
        "primaryAlt4":"#00abee",
        "primaryContrast":"#fff",
        "secondary":"#FFFFFF",
        "secondaryAlt":"#229Fd8",
```

```
    "secondaryAlt2":"#555",
    "secondaryAlt3":"#777",
    "secondaryContrast":"#000",
    "neutralPrimary":"#fff",
    "neutralPrimaryAlt":"#f4f4f4",
    "neutralPrimaryAlt2":"#e3e3e3",
    "neutralPrimaryAlt3":"#c8c8c8",
    "neutralPrimaryContrast":"#000",
    "neutralSecondary":"#fff",
    "neutralSecondaryAlt":"#eaeaea",
    "neutralSecondaryAlt2":"#b7b7b7",
    "neutralSecondaryAlt3":"#acacac",
    "neutralSecondaryContrast":"#000",
    "neutralTertiary":"#b7b7b7",
    "neutralTertiaryAlt":"#c8c8c8",
    "neutralTertiaryAlt2":"#eaeaea",
    "neutralTertiaryAlt3":"#fff",
    "neutralTertiaryContrast":"#222",
    "danger":"#bb2124",
    "success":"#2b3",
    "warning":"#f0ad4e",
    "info":"#5bc0de",
    "dangerContrast":"#fff",
    "successContrast":"#fff",
    "warningContrast":"#fff",
    "infoContrast":"#fff",
    "kpiGood":"#4fb443",
    "kpiBad":"#de061a",
    "kpiNeutral":"#d9b42c",
    "kpiNone":"#333",
    "kpiGoodContrast":"#fff",
    "kpiBadContrast":"#fff",
    "kpiNeutralContrast":"#fff",
    "kpiNoneContrast":"#fff"
},
"theme":{
    "dataPoints":[
        "#0072c6",
        "#f68c1f",
        "#269657",
        "#dd5900",
        "#5b3573",
        "#22bdef",
```

```
            "#b4009e",
            "#008274",
            "#fdc336",
            "#ea3c00",
            "#00188f",
            "#9f9f9f"
        ],

        "good":"#85ba00",
        "bad":"#e90000",
        "neutral":"#edb327",
        "none":"#333",

        "background":"#fff",
        "foreground":"#222",
        "mapBase":"#00aeef",
        "panelBackground":"#f6f6f6",
        "panelForeground":"#222",
        "panelAccent":"#00aeef",
        "tableAccent":"#00aeef",

        "altBackground":"#f6f6f6",
        "altForeground":"#000",
        "altMapBase":"#f68c1f",
        "altPanelBackground":"#235378",
        "altPanelForeground":"#fff",
        "altPanelAccent":"#fdc336",
        "altTableAccent":"#fdc336"
    }
}
```

You will notice the color definitions form five groups: primary, secondary, neutral, danger/success/warning/info, and KPI. Neutral is further divided into neutral primary, neutral secondary, and neutral tertiary.

These color settings control the colors of the web portal, as follows:

- **Primary** Button and hover colors
- **Secondary** Title bar, search bar, page menu (left side of the screen), and text on those items
- **Neutral Primary** Home and report background colors
- **Neutral Secondary** Settings menu, text box background, and folder options background
- **Neutral Tertiary** Site settings background
- **Danger/Success/Warning/Info** Colors for these dialog boxes
- **KPI** Colors for the good, neutral, and bad states of key performance indicators

Logo File

When a logo file is included in the brand package, it takes the place of the name in the upper-left corner of the web portal pages. The logo file must be in the PNG or JPG format. The logo file should be approximately 60 pixels high and 290 pixels wide.

Applying a Brand Package

To apply a brand package, go to the “Site settings” page using the “Site settings” menu option, shown earlier in Figure 16-17. Select the Branding tab on the left side of the page. The Branding tab of the “Site settings” page appears, as shown in Figure 16-24.

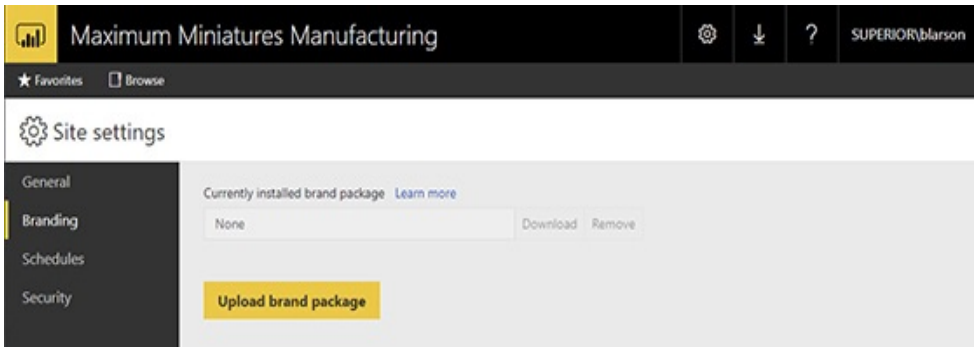

Figure 16-24 *The Branding tab of the "Site settings" page*

To load the brand package, click the "Upload brand package" button. A file open dialog box appears. Navigate to the folder where your brand package ZIP file is saved. Select the brand package ZIP file and click Open. The branding will be applied. Figure 16-25 shows the Power BI Report Server with a brand package applied.

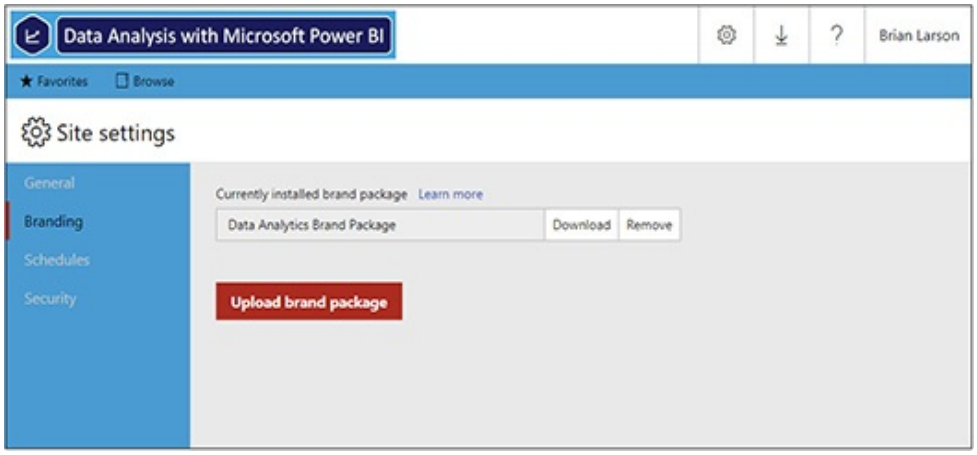

Figure 16-25 *The Power BI Report Server with a brand package applied*

If you ever lose the current brand package ZIP file, a new brand package can be created by clicking the Download button on the Branding tab of the "Site settings" page. Use the Remove button on that page to remove the brand package and

return to the default settings.

POWERED UP AND READY TO GO

You are now powered up and ready. However, Power BI continues to grow and change monthly. Remember to go to

www.teamscs.com/powerbi

to stay up-to-date on the latest enhancements.

By staying current and using all of the tools in the Power BI toolbox, you can use your data to take your organization wherever it needs to go!