Clients First®
BUSINESS SOLUTIONS
Microsoft Dynamics AX Practice

Microsoft

POWER BI

Factsheet

EMPOWERING
BUSINESS

Microsoft

Power BI Security

Summary: Power BI is an online software service (*SaaS*, or Software as a Service) offering from Microsoft that lets you easily and quickly create self-service Business Intelligence dashboards, reports, datasets, and visualizations. With Power BI, you can connect to many different data sources, combine and shape data from those connections, then create reports and dashboards that can be shared with others.

Writer: David Iseminger

Technical Reviewers: Pedram Rezaei, Cristian Petculescu, Haydn Richardson, Adam Wilson, Siva Harinath

Published: February 2016

Applies to: Power BI, Power BI Desktop

Introduction

Power BI is an online software service (*SaaS*, or Software as a Service) offering from Microsoft that lets you easily and quickly create self-service Business Intelligence dashboards, reports, datasets, and visualizations. With Power BI, you can connect to many different data sources, combine and shape data from those connections, then create reports and dashboards that can be shared with others.

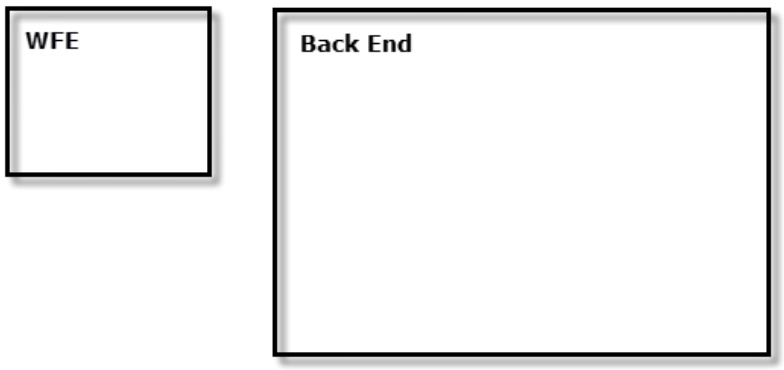
The Power BI service is governed by the Microsoft Online Services Terms, and the Microsoft Enterprise Privacy Statement. For the location of data processing, please refer to the Location of Data Processing terms in the Microsoft Online Services Terms. Power BI is excluded from the Office 365 Trust Center and the Azure Trust Center, as described here. The Power BI team is working hard to bring its customers the latest innovations and productivity. Power BI is currently in Tier A of the Office 365 Compliance Framework, and is working toward moving to Tier D of the Office 365 Compliance Framework over time.

This article describes Power BI security by providing an explanation of the Power BI architecture, then explaining how users authenticate to Power BI and data connections are established, and then describing how Power BI stores and moves data through the service. The last section is dedicated to security-related questions, with answers provided for each.

Power BI Architecture

The **Power BI** service is built on **Azure**, which is Microsoft's cloud computing platform. Power BI is currently deployed in 16 datacenters – there are eight active deployments made available to customers in the regions served by those datacenters, and eight passive deployments that serve as backups for each active deployment.

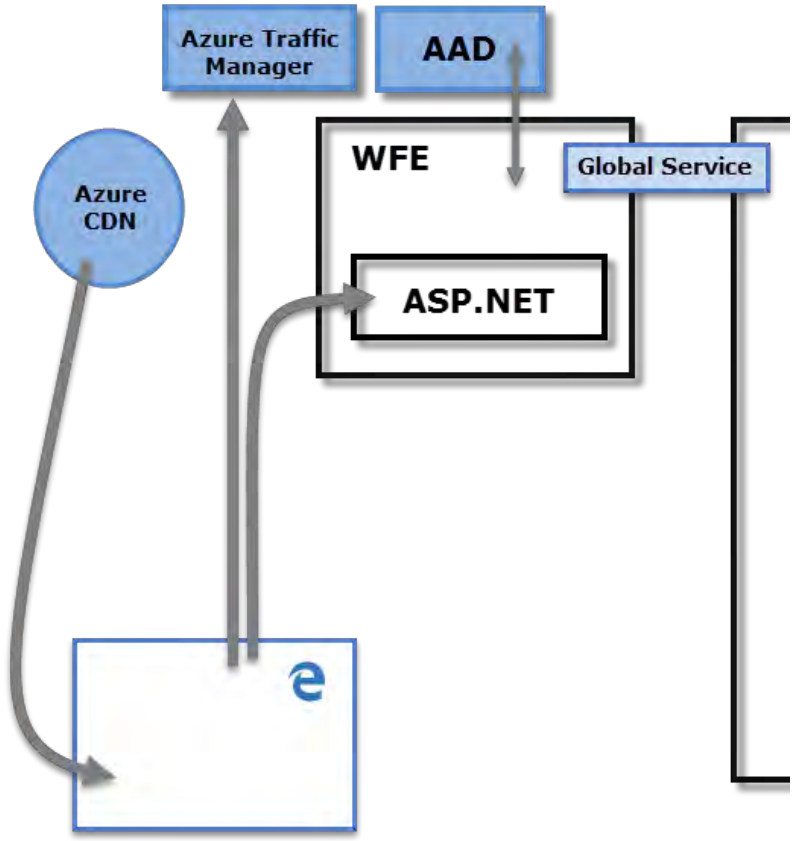
Each Power BI deployment consists of two clusters – a Web Front End (**WFE**) cluster, and a **Back End** cluster. These two clusters are shown in the following image, and provide the backdrop for the rest of this article.



Power BI uses Azure Active Directory (**AAD**) for account authentication and management. Power BI also uses the **Azure Traffic Manager (ATM)** to direct user traffic to the nearest datacenter, determined by the DNS record of the client attempting to connect, for the authentication process and to download static content and files. Power BI uses the **Azure Content Delivery Network (CDN)** to efficiently distribute the necessary static content and files to users based on geographical locale.

The WFE Cluster

The **WFE** cluster manages the initial connection and authentication process for Power BI, using AAD to authenticate clients and provide tokens for subsequent client connections to the Power BI service.

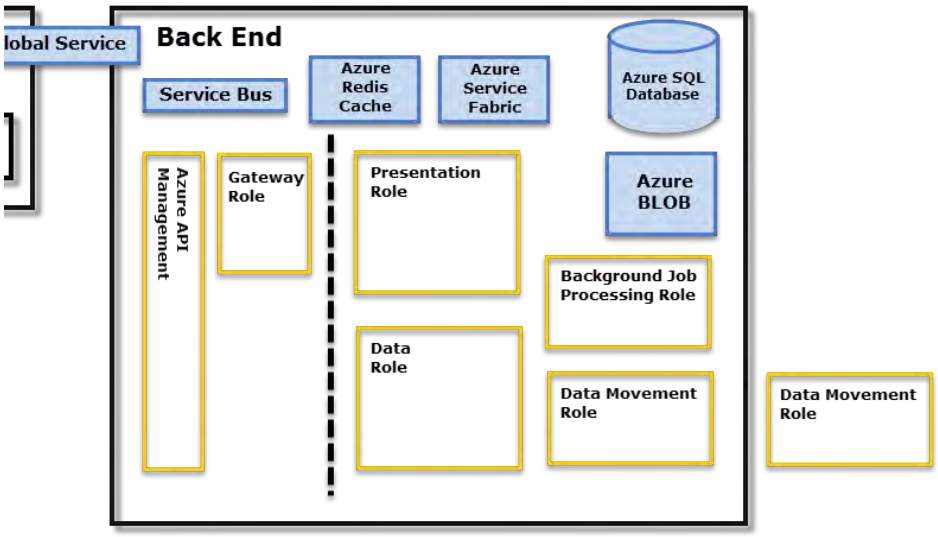


When users attempt to connect to the Power BI service, the client's DNS service may communicate with the **Azure Traffic Manager** to find the nearest datacenter with a Power BI deployment. For more information about this process, see [Performance traffic routing method for Azure Traffic Manager].

The WFE cluster nearest to the user manages the login and authentication sequence (described later in this article), and provides an AAD token to the user once authentication is successful. The ASP.NET component within the WFE cluster parses the request to determine which organization the user belongs to, and then consults the Power BI **Global Service**. The Global Service is a single Azure Table shared among all worldwide WFE and Back End clusters that maps users and customer organizations to the datacenter that houses their Power BI tenant. The WFE specifies to the browser which Back End cluster houses the organization's tenant. Once a user is authenticated, subsequent client interactions occur with the Back End cluster directly, without the WFE being an intermediator for those requests.

The Power BI Back End Cluster

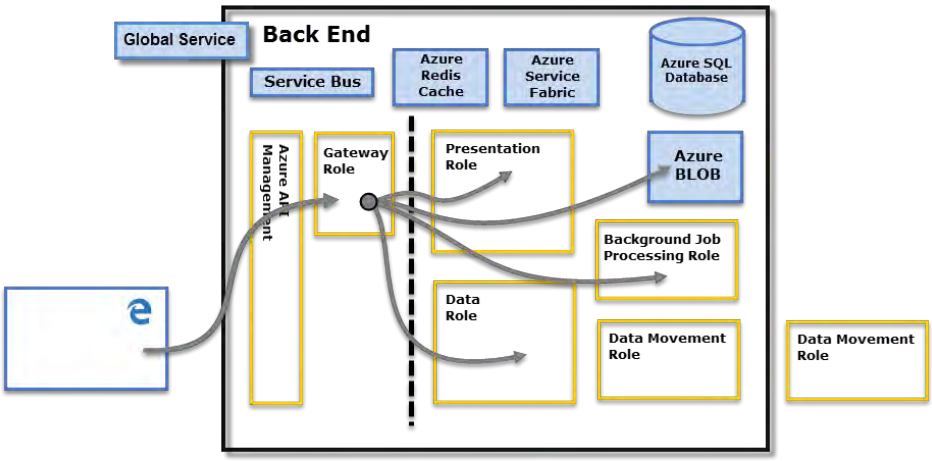
The **Back End** cluster is how authenticated clients interact with the Power BI service. The **Back End** cluster manages visualizations, user dashboards, datasets, reports, data storage, data connections, data refresh, and other aspects of interacting with the Power BI service.



The **Gateway Role** acts as a gateway between user requests and the Power BI service. Users do not interact directly with any roles other than the Gateway Role. **Azure API Management** will eventually handle the Gateway Role.

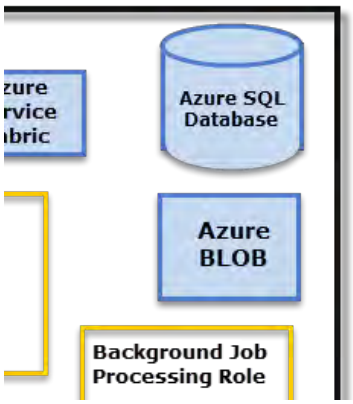
Important: It is imperative to note that *only* Azure API Management (**APIM**) and Gateway (**GW**) roles are accessible through the public Internet. They provide authentication, authorization, DDoS protection, Throttling, Load Balancing, Routing, and other capabilities.

The dotted line in the **Back End** cluster image, above, clarifies the boundary between the only two roles that are accessible by users (left of the dotted line), and roles that are only accessible by the system. When an authenticated user connects to the Power BI Service, the connection and any request by the client is accepted and managed by the **Gateway Role** (eventually to be handled by **Azure API Management**), which then interacts on the user's behalf with the rest of the Power BI Service. For example, when a client attempts to view a dashboard, the **Gateway Role** accepts that request then separately sends a request to the **Presentation Role** to retrieve the data needed by the browser to render the dashboard.



Data Storage Architecture

Power BI uses two primary repositories for storing and managing data: data that is uploaded from users is typically sent to **Azure BLOB** storage, and all metadata as well as artifacts for the system itself are stored in **Azure SQL Database**.



For example, when a user imports an Excel workbook into the Power BI service, an in-memory Analysis Services tabular database is created, and the data is stored in-memory for approximately an hour (or until memory pressure occurs on the system). The data is also sent to **Azure BLOB** storage.

Metadata about a user's Power BI subscription, such as dashboards, reports, recent data sources, workspaces, organizational information, tenant information, and other metadata about the system is stored and updated in **Azure SQL Database**. Both the Azure BLOB storage and the Azure SQL Database are generally unencrypted. More information about the process of loading, storing, and moving data is described in the **Data Storage and Movement** section.

Tenant Creation

A tenant is a dedicated instance of the Azure AD service that an organization receives and owns when it signs up for a Microsoft cloud service such as Azure, Microsoft Intune, Power BI, or Office 365. Each Azure AD tenant is distinct and separate from other Azure AD tenants.

A tenant houses the users in a company and the information about them - their passwords, user profile data, permissions, and so on. It also contains groups, applications, and other information pertaining to an organization and its security. For more information, see What is an Azure AD tenant.

In Power BI, a tenant is created in the datacenter deemed closest, at the time when the Power BI service is initially provisioned. The Power BI tenant does not move from that datacenter location today, but the Power BI development team is working on a scenario to allow tenant administrators to move their subscription and data from one region to another.

For example, if the IT manager for Contoso Inc. decided she wanted a Power BI subscription for all employees of Contoso, and she initiated the creation of that subscription from Seattle, Washington (in the western United States), Power BI would create the Contoso Power BI tenant in the West US datacenter (the closest datacenter to Seattle). No matter where other employees reside – whether in

Europe, Asia, Florida, or Australia – every employee would connect to the Power BI service cluster housed in the West US datacenter, because that is where the cluster resides.

For another example, if the IT manager happened to be on a business trip in Asia when she initially signed Contoso up for a Power BI subscription, the tenant would be created in the nearest datacenter (in this case, Southeast Asia). After that, all subsequent connections by Contoso employees to their Power BI service would go to Southeast Asia. Once created, you cannot move tenants from one datacenter to another.

Datacenters and Locales

Power BI is offered in certain regions, based on where Power BI clusters are deployed in regional datacenters. Microsoft plans to expand its Power BI infrastructure into additional datacenters.

The following links provide additional information about Azure datacenters.

- Azure Regions – information about Azure's global presence and locations
- Azure Services, by region – a complete listing of Azure services (both infrastructure services and platform services) available from Microsoft in each region.

Currently, the Power BI service is available in the following regions, serviced by the following datacenters:

- West US
- North Central US
- South Central US
- East US 2
- West Europe
- North Europe
- Southeast Asia
- Brazil

Note: Although Power BI tenants remain in the datacenter in which they were created, it currently is not possible to guarantee that data processing initiated in one region will stay in that region.

User Authentication

User authentication to the Power BI service consists of a series of requests, responses, and redirects between the user's browser and the Power BI service or the Azure services used by Power BI. That sequence describes the process of user authentication in Power BI. For more information about options for an organization's user authentication models (sign-in models), see Choosing a sign-in model for Office 365.

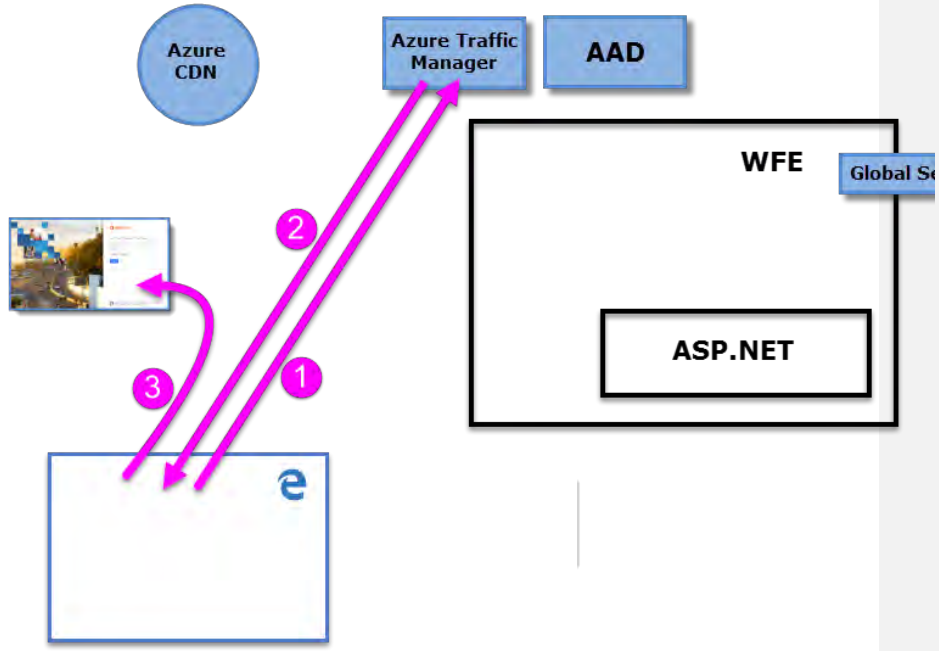
Authentication Sequence

The user authentication sequence for the Power BI service occurs as described in the following steps, which are illustrated in the following images.

1. A user initiates a connection to the Power BI service from a browser, either by typing in the Power BI address in the address bar (such as https://app.powerbi.com) or by selecting Sign In from the Power BI landing page (https://powerbi.microsoft.com). The connection is established using HTTPS, and all subsequent communication between the browser and the Power BI service uses HTTPS. The request is sent to the **Azure Traffic Manager**.

2. The **Azure Traffic Manager** checks the user's DNS record to determine the nearest datacenter where Power BI is deployed, and responds to the DNS with the IP address of the WFE cluster to which the user should be sent.

3. WFE then redirects the user to Microsoft Online Services login page.



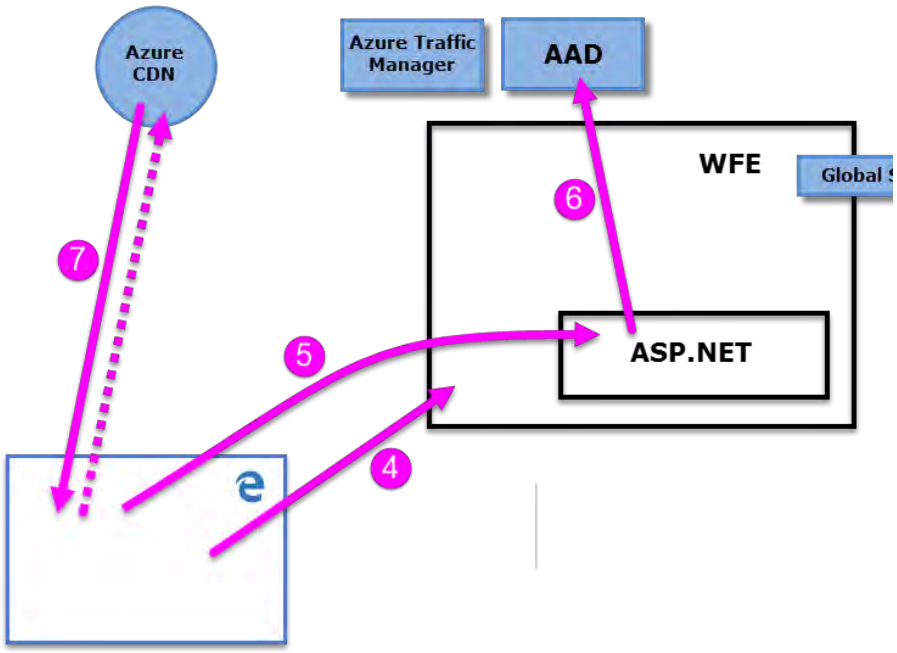
4. Once the user is authenticated, the login page redirects the user to the previously determined nearest Power BI service **WFE cluster**.

5. The browser submits a cookie that was obtained from the successful login to Microsoft Online Services, which is inspected by the **ASP.NET service** inside the **WFE cluster**.

6. The WFE cluster checks with the **Azure Active Directory** (**AAD**) service to authenticate the user's Power BI service subscription, and to obtain an AAD security token. When AAD returns successful authentication of the user and returns an AAD security token, the WFE cluster consults the **Global Service,** which maintains a list of tenants and their Power BI Back End cluster locations, and determines which Power BI service cluster contains the user's tenant. The WFE cluster then directs the user to the Power BI cluster where its tenant resides, and returns a collection of items to the user's browser:

    - The **AAD security token**
    - **Session information**
    - The web address of the **Back End** cluster the user can communicate and interact with

7. The user's browser then contacts the specified Azure CDN, or for some of the files the WFE, to download the collection of specified common files necessary to enable the browser's interaction with the Power BI service. The browser page then includes the AAD token, session information, the location of the associated Back End cluster, and the collection of files downloaded from the Azure CDN and WFE cluster, for the duration of the Power BI service browser session.



Once those items are complete, the browser initiates contact with the specified Back End cluster and the user's interaction with the Power BI service commences. From that point forward, all calls to the Power BI service are with the specified Back End cluster, and all calls include the user's AAD token.

Data Storage and Movement

In the Power BI service, data is either *at rest* (data available to a Power BI user that is not currently being acted upon), or it is *in process* (for example: queries being run, data connections and models being acted upon, data and/or models being uploaded into the Power BI service, and other actions that users or the Power BI service may take on data that is actively being accessed or updated). Data that is in process is referred to as *data in process*.

The Power BI service also manages data differently based on whether the data is accessed with a **Direct Query**, or is *not* accessed with a Direct Query. So there are two categories of user data for Power BI: data that is accessed by Direct Query, and data which is not accessed by Direct Query.

A **Direct Query** is a query for which a Power BI user's query has been translated from Microsoft's Data Analysis Expressions (DAX) language – which is the language used by Power BI and other Microsoft

products to create queries – in the data source's native data language (such as SQL, or other native database languages). The data associated with a Direct Query is stored by reference only, which means source data is not stored in Power BI when the Direct Query is not active (except for visualization data used to display dashboards and reports, as described in the *Data in process (data movement)* section, below). Rather, references to Direct Query data are stored which allow access to that data when the Direct Query is run. A Direct Query contains all the necessary information to execute the query, including the connection string and the credentials used to access the data sources, which allow the Direct Query to connect to the included data sources for automatic refresh. With a Direct Query, underlying data model information is incorporated into the Direct Query.

A query that does **not** use Direct Query consist of a collection of DAX queries that are *not* directly translated to the native language of any underlying data source. Non-Direct Query queries do not include credentials for the underlying data, and the underlying data is loaded into the Power BI service unless it is on-premises data accessed through a Gateway, in which case the query only stores references to on-premises data.

The distinction between a Direct Query and other queries determines how the Power BI service handles the data at rest, and whether the query itself is encrypted. The following sections describe data at rest and in movement, and explain the encryption, location, and process for handling data.

Data at rest

When data is at rest, the Power BI service stores datasets, reports, and dashboard tiles in the manner described in the following sub-sections. ETL stands for Extract, Transform and Load in the following sections.

Datasets

1. Metadata (tables, columns, measures, calculations, connection strings, etc.)
    a. Analysis Services on-premises – nothing is stored
    b. Direct Query[1] – encrypted in Azure BLOB storage. The encryption and access to Azure BLOB keys are stored in a separate location in the Power BI service.
    c. ETL and pushed data (non-Direct Query, non-Analysis Services on-premises) – not encrypted (clear text) in Azure BLOB storage. The access key to the Azure BLOB is stored in a separate location in the Power BI service[2].
2. Credentials to the original data sources
    a. Analysis Services on-premises – nothing is stored
    b. Direct query – stored in the connection string, as described in 1.b., above
    c. Pushed data – none (not applicable)
    d. ETL

[1] Direct query datasets are currently created only using the content packs for connecting Azure SQL Database, Azure SQL Data Warehouse and Spark on Azure HDInsight. They cannot be created using Excel or Power BI Desktop.
[2] Employees' access to the keys is possible but is regulated by internal policies.

i. From **Salesforce** or **OneDrive** – the refresh tokens are stored encrypted in the SQL Azure Database of the Power BI service. The credentials to access the database are stored in a separate location in the Power BI service[3].
ii. Otherwise:
- If the dataset is set for refresh, the credentials are stored encrypted in a SQL Azure Database used by the Data Movement Role. The credentials to the database are stored in a separate location within the Data Movement Role. The encryption key is stored:
  - In the gateway – for on-premises data sources
  - In the Data Movement Role – for cloud-based data sources
- If the dataset is not set for refresh, there are no credentials stored for the data sources

3. Data
   a. Analysis Services on-premises, and Direct Query – nothing is stored.
   b. ETL and pushed data – not encrypted (clear text) in Azure BLOB storage. The access key to the Azure BLOB is stored, encrypted, in a separate location in the Power BI service.

Reports

1. Metadata (report definition)
   a. Reports can either be Excel for Office 365 reports, or Power BI Desktop (.pbix) reports. The following applies for metadata based on the type of report:
      a. Excel Report metadata is stored encrypted in SQL Azure. Metadata is also stored in Office 365.
      b. Power BI Desktop reports are store encrypted in SQL Azure The credentials to access the database are stored in a separate location in the Power BI service.
2. Static data
   Static data includes artifacts such as background images and custom visuals.
   a. For reports created with Excel for Office 365, nothing is stored.
   b. For Power BI Desktop reports, the static data is stored unencrypted in Azure BLOB storage.
3. Caches
   a. For reports created with Excel for Office 365, nothing is cached.
   b. For Power BI Desktop reports, data for the visuals shown ins cached unencrypted in Azure SQL Database.
4. Original .pbix or .xlsx metadata, when imported into the Power BI service
   a. For Excel for Office 365, nothing is stored in the Power BI service.
   b. For Power BI Desktop reports, the report is stored unencrypted in Azure BLOB storage.

Dashboard Tiles

1. Caches
   a. The data shown (at the base of the visual shown) is stored unencrypted (clear text) in the SQL Azure Database of the service. The credentials to access the database are stored,

[3] Employees' access to the keys is possible but is regulated by internal policies.

encrypted, in a separate location in the Power BI service. This data is refreshed based on the dataset type:

i. For the Analysis Services on-premises – refreshed every time the on-premises dataset (cube) changes
ii. For the Direct Query – refreshed every 15 minutes
iii. For the ETL and pushed data – refreshed every time the dataset changes

Data Transiently Stored on Non-Volatile Devices

The following describes data that is transiently stored on non-volatile devices.

Datasets

1. Metadata (tables, columns, measures, calculations, connection strings, etc.)
    a. Analysis Services on-premises – nothing is stored
    b. Direct Query, ETL and pushed data – stored unencrypted (clear text) on the disk of the compute nodes of the machines of the roles[4].
2. Credentials to the original data sources
    a. Analysis Services on-premises – nothing is stored
    b. Direct Query – stored in the connection string, in encrypted format with the encryption key stored in clear text in the same place (alongside the encrypted information).
    c. Pushed data – none (not applicable)
    d. ETL – none (nothing stored on the compute node nor different than explained in the **Data at Rest** section, above)
3. Data
    a. Analysis Services on-premises, and Direct Query – nothing is stored
    b. ETL and pushed data – unencrypted (clear text) on the disk of the compute nodes of the machines of the roles.

Data in process (data movement)

Data is in process when it is actively being used or accessed by a user. For example, when a user accesses a dataset, revises or modifies a dashboard or report, when refresh occurs, or other data access activities that may occur). When any of those events occur to put data in process, the **Data Role** in the Power BI service creates an in-memory Analysis Services (AS) database and the dataset is loaded into that Analysis Services database. Whether the dataset is based on a Direct Query or not, data loaded in the AS database is unencrypted.

Once data is acted upon, which includes initially loading data into Power BI, the Power BI service caches the visualization data in **Azure SQL Database**, regardless of whether the dataset is based on a Direct Query.

[4] Employees' access using RDP to the compute machines is regulated by internal policies.

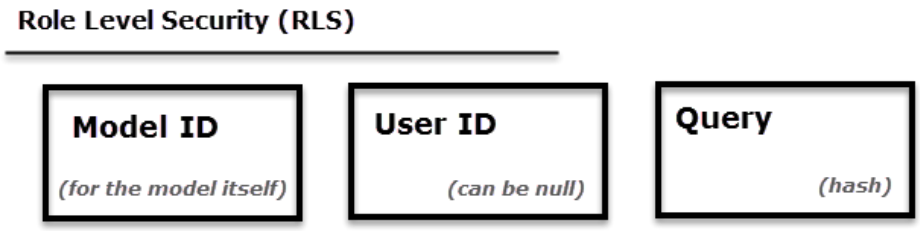
User Authentication to Data Sources

With each data source, a user establishes a connection based on his or her login, and accesses the data with those credentials. Users can then create queries, dashboards, and reports based on the underlying data.

When a user shares queries, dashboards, reports, or any visualization, access to that data and those visualizations is dependent on whether the underlying data sources support Role Level Security (RLS).

If an underlying data source is capable of **Role Level Security (RLS)**, the Power BI service will apply that role level security, and users who do not have sufficient credentials to access the underlying data (which could be a query used in a dashboard, report, or other data artifact) will not see data for which the user does not have sufficient credentials. If a user's access to the underlying data is different from the user who created the dashboard or report, the visualizations and other artifacts will only show data based on the level of access that user has to the data.

If a data source does **not** apply RLS, then the Power BI login credentials are applied to the underlying data source, or if other credentials are supplied during the connection, those supplied credentials are applied. When a user loads data into the Power BI service from non-RLS data sources, the data is stored in Power BI as described in the **Data Storage and Movement** section found in this document. For non-RLS data sources, when data is shared with other users (such as through a dashboard or report) or a refresh of the data occurs, the original credentials are used to access or display the data.



For a quick example to contrast RLS and non-RLS data sources, imagine Sam creates a report and a dashboard, then shares them with Abby and Ralph. If the data sources used in the report and dashboard are from data sources that **do not** support RLS, both Abby and Ralph will be able to see the data that Sam included in the dashboard (which was uploaded into the Power BI service) and both Abby and Ralph can then interact with the data. In contrast, if Sam creates a report and dashboard from data sources that do support RLS, then shares it with Abby and Ralph, when Abby attempts to view the dashboard the following occurs:

1. Since the dashboard is from an RLS data source, the dashboard visualizations will briefly show a "loading" message while the Power BI service queries the data source to retrieve the current dataset specified in the connection string associated with the dashboard's underlying query.

2. The data is accessed and retrieved based on Abby's credentials and role, and only data for which Abby has sufficient authorization is loaded into the dashboard and report.

3. The visualizations in the dashboard and report are displayed based on Abby's role level.

If Ralph were to access the shared dashboard or report, the same sequence occurs based on his role level.

Power BI Security Questions and Answers

The following questions are common security questions and answers for Power BI.

How do users connect to, and gain access to data sources while using Power BI?

Power BI credentials and domain credentials: Users login to Power BI using an email address; when a user attempts to connect to a data resource, Power BI passes the Power BI login email address as credentials. For domain-connected resources (either on-premises or cloud-based), the login email is matched with a *User Principal Name* (UPN) by the directory service to determine whether sufficient credentials exist to allow access. For organizations that use work-based email addresses to login to Power BI (the same email they use to login to work resources, such as *david@contoso.com*), the mapping can occur seamlessly; for organizations that did not use work-based email addresses (such as *david@contoso.onmicrosoft.com*), directory mapping must be established in order to allow access to on-premises resources with Power BI login credentials.

SQL Server Analysis Services and Power BI: For organizations that use on-premises SQL Server Analysis Services, Power BI offers the Power BI Analysis Services Connector (which is a **Gateway**, as referenced in previous sections). The Power BI Analysis Services Connector can enforce role- and row-level security on data sources (RLS). For more information on RLS, see **User Authentication to Data Sources** earlier in this document. You can also read an in-depth treatment of Power BI Analysis Services Connector, get configuration guidance, and get answers to common questions, or download the Power BI Analysis Services Connector.

Non-domain connections: For data connections that are not domain-joined and not capable of Role Level Security (RLS), the user must provide credentials during the connection sequence, which Power BI then passes to the data source to establish the connection. If permissions are sufficient, data is loaded from the data source into the Power BI service.

How is data transferred to Power BI?

All data requested and transmitted by Power BI is encrypted in transit using HTTPS to connect from the data source to the Power BI service. A secure connection is established with the data provider, and only once that connection is established will data traverse the network.

How does Power BI cache report, dashboard, or model data, and is it secure?

When a data source is accessed, the Power BI service follows the process outlined in the **Data Storage and Movement** section earlier in this document.

What about role-based security, sharing reports or dashboards, and data connections? How does that work in terms of data access, dashboard viewing, report access or refresh?

For **non-Role Level Security (RLS)** enabled data sources, if a dashboard, report, or data model is shared with other users through Power BI, the data is then available for users with whom it is shared to view and interact with. Power BI ***does not*** re-authenticate users against the original source of the data; once data is uploaded into Power BI, the user who authenticated against the source data is responsible for managing which other users and groups can view the data.

When data connections are made to an **RLS**-capable data source, such as an Analysis Services data source, only dashboard data is cached in Power BI. Each time a report or dataset is viewed or accessed in Power BI that uses data from the RLS-capable data source, the Power BI service accesses the data source to get data based on the user's credentials, and if sufficient permissions exist, the data is loaded into the report or data model for that user. If authentication fails, the user will see an error.

For more information, see the **User Authentication to Data Sources** section earlier in this document.

Our users connect to the same data sources all the time, some of which require credentials that differ from their domain credentials. How can they avoid having to input these credentials each time they make a data connection?

Power BI offers the Power BI Personal Gateway, which is a feature that lets users create credentials for multiple different data sources, then automatically use those credentials when subsequently accessing each of those data sources. For more information, see Power BI Personal Gateway.

How do Power BI Groups work?

Power BI Groups allow users to quickly and easily share their dashboards, reports, and data models with established teams. For example, if you have a Power BI Group that includes everyone in your immediate team, you can easily share your most recent report with everyone on your team by selecting the Group from within Power BI. Power BI Groups are equivalent to Office 365 Universal Groups (which you can learn about, create, and manage), and use the same authentication mechanisms used in Azure Active Directory to secure data. You can create groups in Power BI or create a Universal Group in Office 365 admin center; either has the same result for group creation in Power BI.

Note that data shared with Power BI Groups follows the same security consideration as any shared data in Power BI. For **non-RLS** data sources Power BI does ***not*** re-authenticate users against the original source of data, and once data is uploaded into Power BI, the user who authenticated against the source data is responsible for managing which other users and groups can view the data. For more information, see the **User Authentication to Data Sources** section earlier in this document.

You can get more information about Groups in Power BI.

Which ports are used by Analysis Services Connector and Personal Gateway? Are there any domain names that need to be allowed for connectivity purposes?

For Power BI, the Analysis Services Connector and Personal Gateway use the same ports. All service connections are outbound (from the on-premises listening server), initiated by Service Bus, so there's no need to open incoming ports on the on-premises server.

The following steps outline the connection process, where the listener is the on-premises server on which the Analysis Services Connector or Personal Gateway is running:

1. Upon receiving a connection request from Service Bus, the listener attempts to connect to Service Bus on port **5672**.
2. If connection on port **5672** is not successful, the listener attempts to connect on port **443**.
3. Once the connection is established, the listener will attempt to rendezvous using ports **9350** through **9354**.
4. If rendezvous fails on the **9350 – 9354** port range, then a rendezvous on port **443** is attempted.

As such, the only port requirement for the Analysis Services Connector and Personal Gateway is port 443, however the other ports listed in the above process will be attempted first, before falling back to port 443.

During the process, the listener will attempt to communicate with domains necessary to establish a secure connection with the Power BI service. In cases where domain connections are

blocked unless explicitly allowed, the following domains may need to be added to the approved connection list:

Domain Name	Port
*.powerbi.com	443
*.analysis.windows.net	443
*.core.windows.net	443
*.login.windows.net	443
*.servicebus.windows.net	443, 9350, 9351, 9352, 9353

Conclusion

The Power BI service architecture is based on two clusters – the Web Front End (WFE) cluster and the Back End cluster. The WFE cluster is responsible for initial connection and authentication to the Power BI service, and once authenticated, the Back End handles all subsequent user interactions. Power BI uses Azure Active Directory (AAD) to store and manage user identities, and manages the storage of data and metadata using Azure BLOB and Azure SQL Database, respectively.

Data storage and data processing in Power BI differs based on whether data is accessed using a Direct Query, and is also dependent on whether data sources are in the cloud or on-premises. Power BI is also capable of enforcing Role Level Security (RLS) and interacts with Gateways that provide access to on-premises data.

Feedback and Suggestions

We appreciate your feedback. We're interested in hearing any suggestions you have for improvement, additions, or clarifications to this whitepaper, or other content related to Power BI. Please send your suggestions to pbidocfeedback@microsoft.com.

Additional Resources

For additional information on Power BI, see the following resources.

- Power BI Analysis Services Connector in-depth
- Power BI Personal Gateway
- Groups in Power BI
- Getting Started with Power BI Desktop
- Power BI REST API - Overview
- Real-time Power BI analytics (application development)
- Microsoft Power BI API reference

Why Microsoft?

Microsoft Dynamics is the Microsoft's business solution for enterprises that enables people to make smarter decisions faster with access to real-time insights and intelligence on nearly any device, anywhere. It enables business to redesign their business processes faster so they can innovate and get quick time to value to stay ahead of the competition. It gives businesses the flexibility to grow at their pace through the choice and flexibility of the cloud, allowing them to scale their operations globally to meet business needs.

Microsoft POWER BI Factsheet

EMPOWERING
BUSINESS

Microsoft

Clients First | Phone 800.331.8382 | Address 1000 Ballpark Way, Suite 315, Arlington, TX 76011 | http://clientsfirst-ax.com