



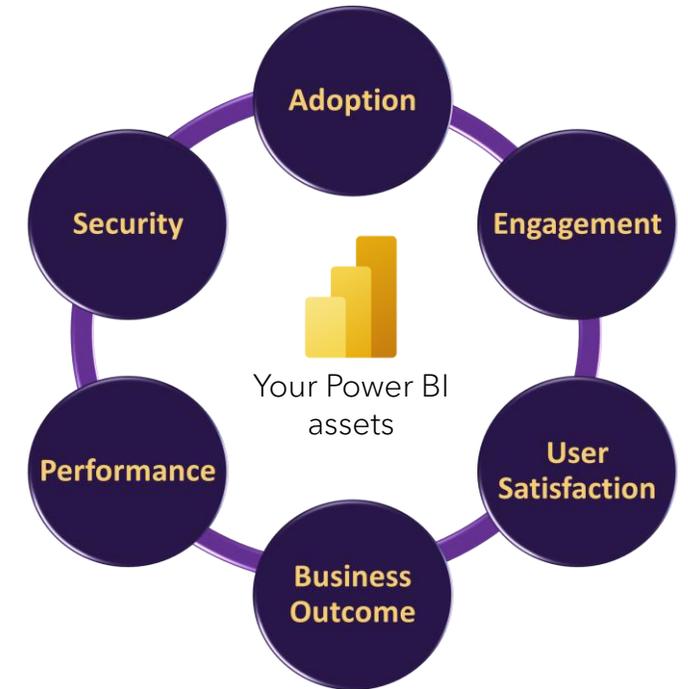
# BI Pixie: Architecture & Security

Gil Raviv, DataChant  
[gilra@datachant.com](mailto:gilra@datachant.com)



# WHAT IS BI PIXIE?

- [BI Pixie](#) empowers BI teams to deliver with excellence and maximize their ROI in Power BI. It collects and analyzes unique usage data to support decision-making and optimizations across 6 areas: Adoption, Engagement, User Satisfaction, Business Outcome, Performance and Security.
- The data never leaves your organization
- It can track engagement from any platform (Power BI service, Power BI desktop, Power BI mobile, Power BI embedded (app-owns-data or user-owns-data), Publish to web, Power BI report server)
- No use of custom visuals. The solution uses built-in Power BI building blocks
- Deployment is super-easy and doesn't require IT skills
- User guide is available [here](#)

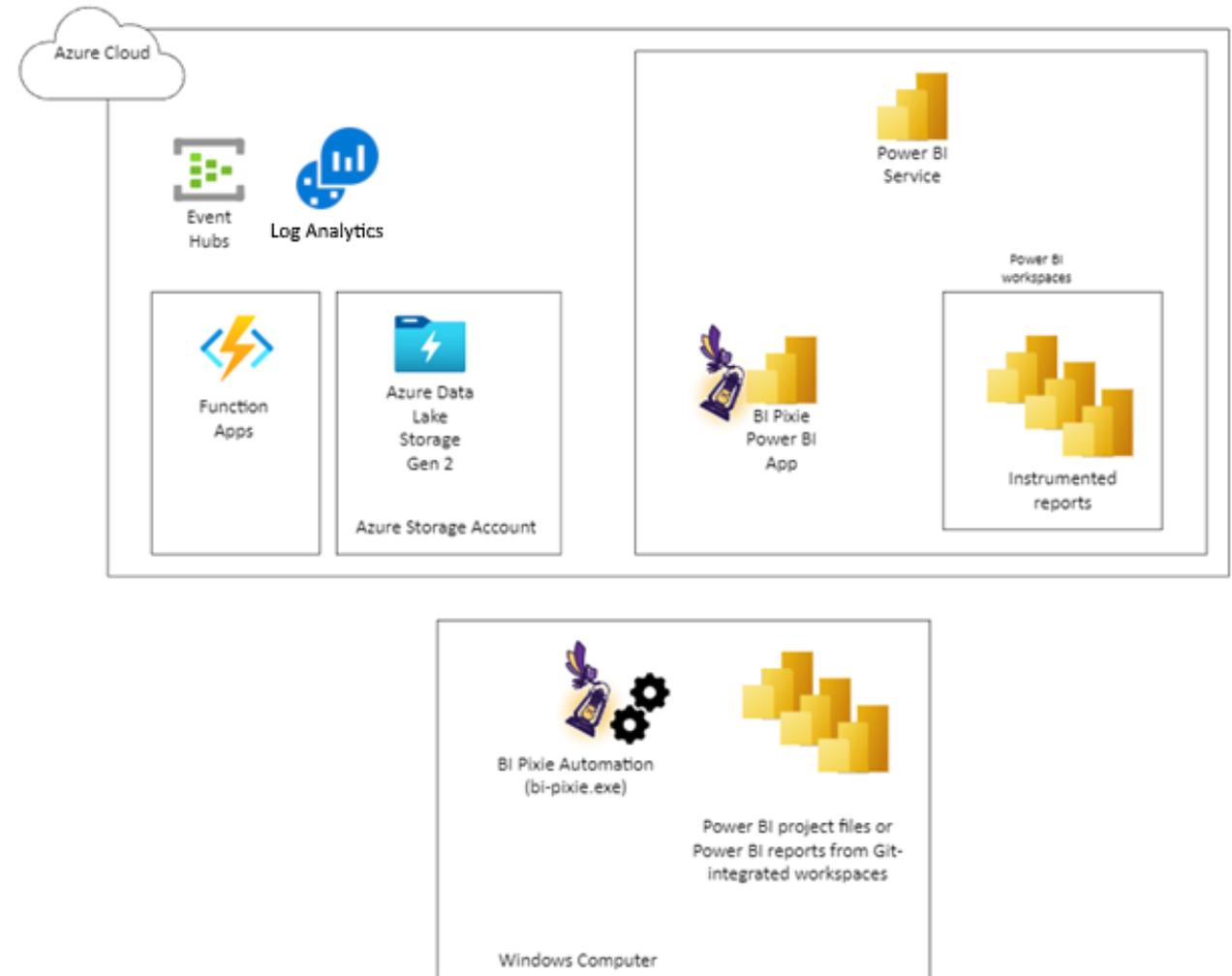




# SERVERLESS ARCHITECTURE

BI Pixie includes 5 components that can be deployed in your Azure account

- BI Pixie Instrumentation executable (AKA "BI Pixie Automation")
- A web app component using Azure Function App
- Event Hubs to stream the events from the web app to storage
- Storage of events using ADLS gen2 with [hierarchical namespace](#)
- Analytics and data visualization with [Power BI App](#)





# A SERVERLESS ARCHITECTURE THAT CAN SCALE



BI Pixie runs on your reports and instrument them to track engagement on any platform. The instrumentation can be added to your current ALM processes.



Power BI

Your BI audience



Azure Function App

Default: TLS 1.2  
Can be set to TLS1.3

Azure Deployment



Azure Storage Account (ADLS gen2)

Data at rest  
Service-Side Encryption  
256-bit AES, FIPS  
140-2 compliant



Power BI App





# SERVERLESS ARCHITECTURE ON AZURE

	Single-Mode (One Function App) Architecture	Dual-Mode (Two Function Apps) Architecture
Availability	Azure Marketplace - Solution template or managed application	Functions can be shared with you, and we can help you deploy it
Main components	1 function app 1 storage account 1 Event Hub	2 function apps 1-2 storage account 1 Event Hub
Data movement	<b>Unified function app:</b> Web trigger → Event Hub Event Hub Trigger → Blob Timer → Log Analytics Timer → Power BI Admin APIs Timer → Blob	<b>External function app:</b> Web trigger → Event Hub  <b>Internal function app:</b> Event Hub Trigger → Blob Timer → Log Analytics Timer → Power BI Admin APIs Timer → Blob
Advantages	Can meet higher security bar with Event Hub that decouples function app and storage	Can meet higher security bars with separation of networks
	Supports real-time use-cases	

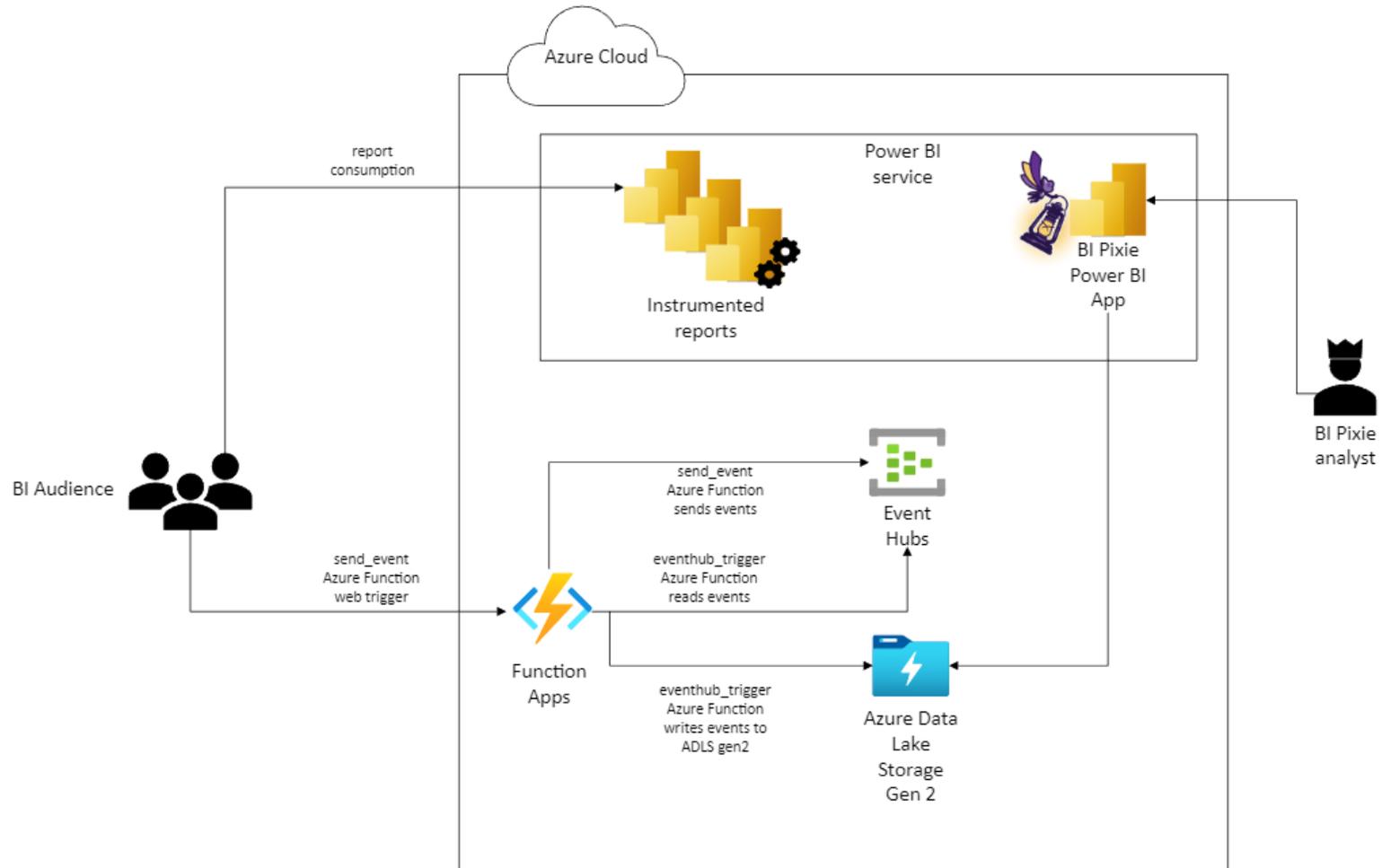


# DATA SOURCES, FUNCTIONS AND FEATURES

Data source	Function used	Features
BI Pixie Instrumentation Web events	send_events	<ul style="list-style-type: none"> <li>• Instrumented activities</li> <li>• Heatmaps: Clicks, Performance, Exports from visuals</li> <li>• Feedback controls</li> <li>• Embedded surveys</li> <li>• Data auditing</li> <li>• RLS auditing</li> </ul>
Azure Log Analytics	log_analytics	<ul style="list-style-type: none"> <li>• Performance</li> <li>• Exports from visuals</li> <li>• RLS auditing</li> </ul>
Power BI Activity Logs	pbi_activities	<ul style="list-style-type: none"> <li>• Consumers and Producers</li> <li>• Paginated reports</li> </ul>
Power BI Tenant Scans	pbi_scans	<ul style="list-style-type: none"> <li>• Data sources lineage</li> <li>• Endorsed semantic models and reports</li> </ul>

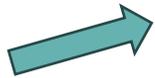
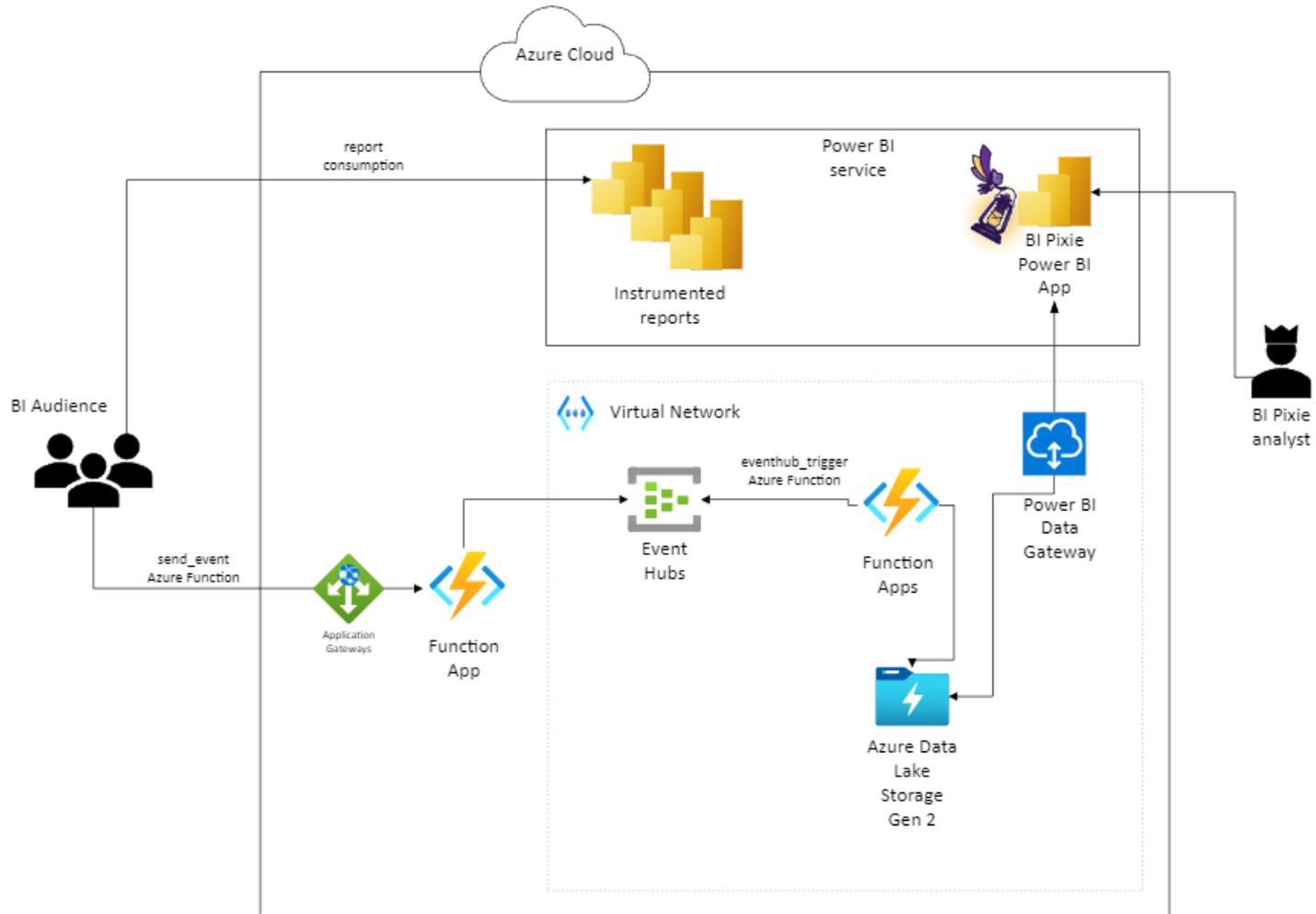


# SINGLE-MODE ARCHITECTURE





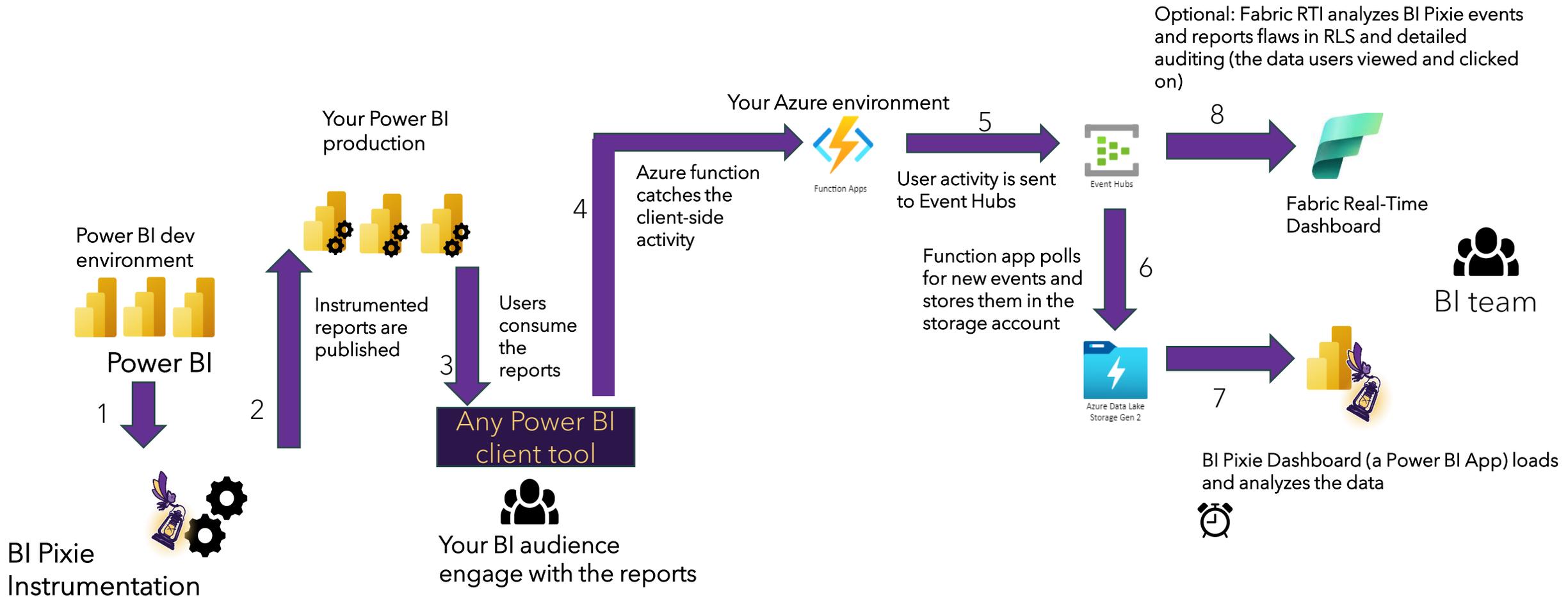
# DUAL-MODE DEPLOYMENT + VNET



You can use your application gateway to prevent DDoS and hide the function app behind a private or service endpoint



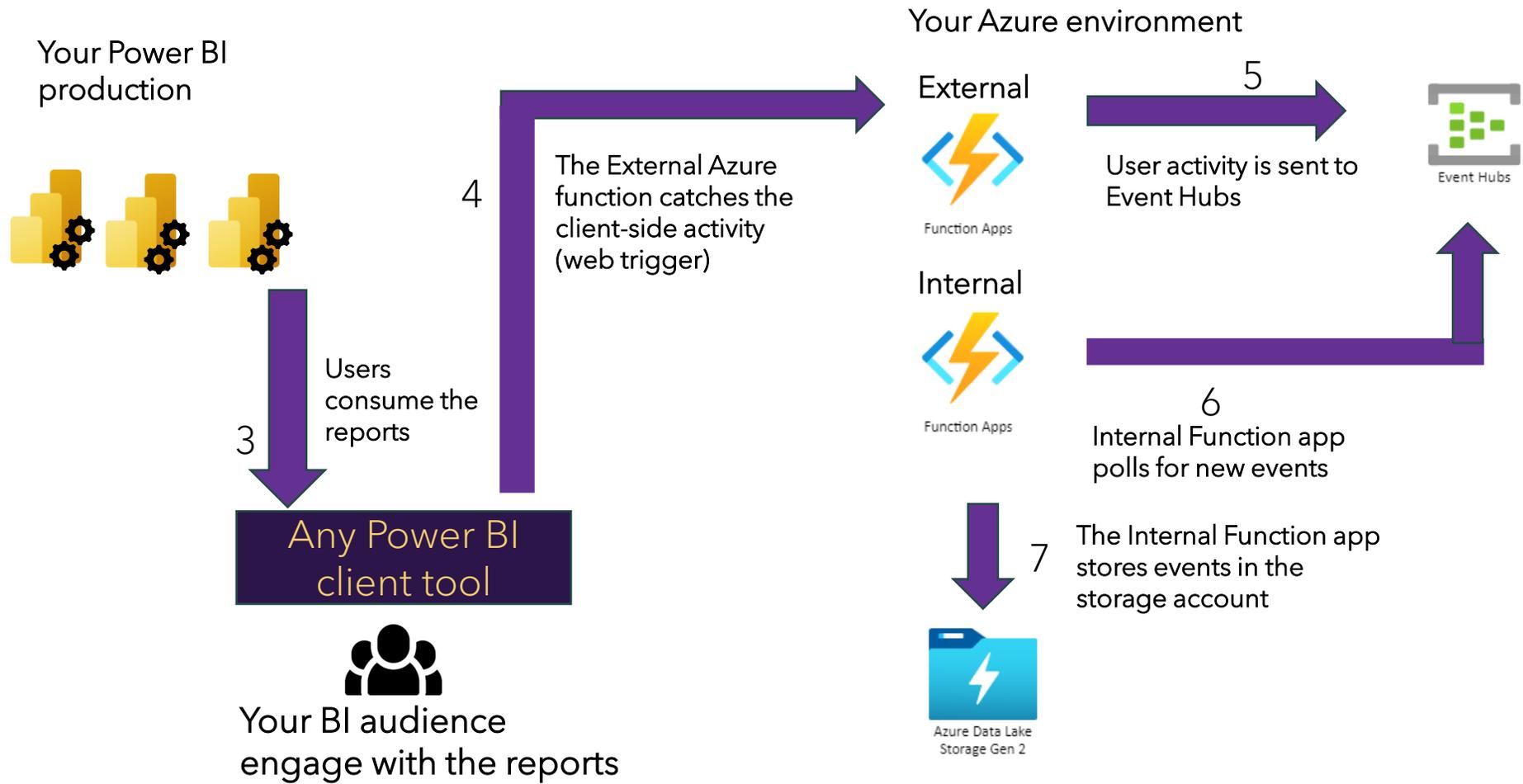
# BI PIXIE WORKFLOW IN SINGLE-MODE ARCHITECTURE



- All components can be installed and ready to be used in less than an hour.
- All data stays on your Azure account.



# BI PIXIE WORKFLOW IN ADVANCED ARCHITECTURE

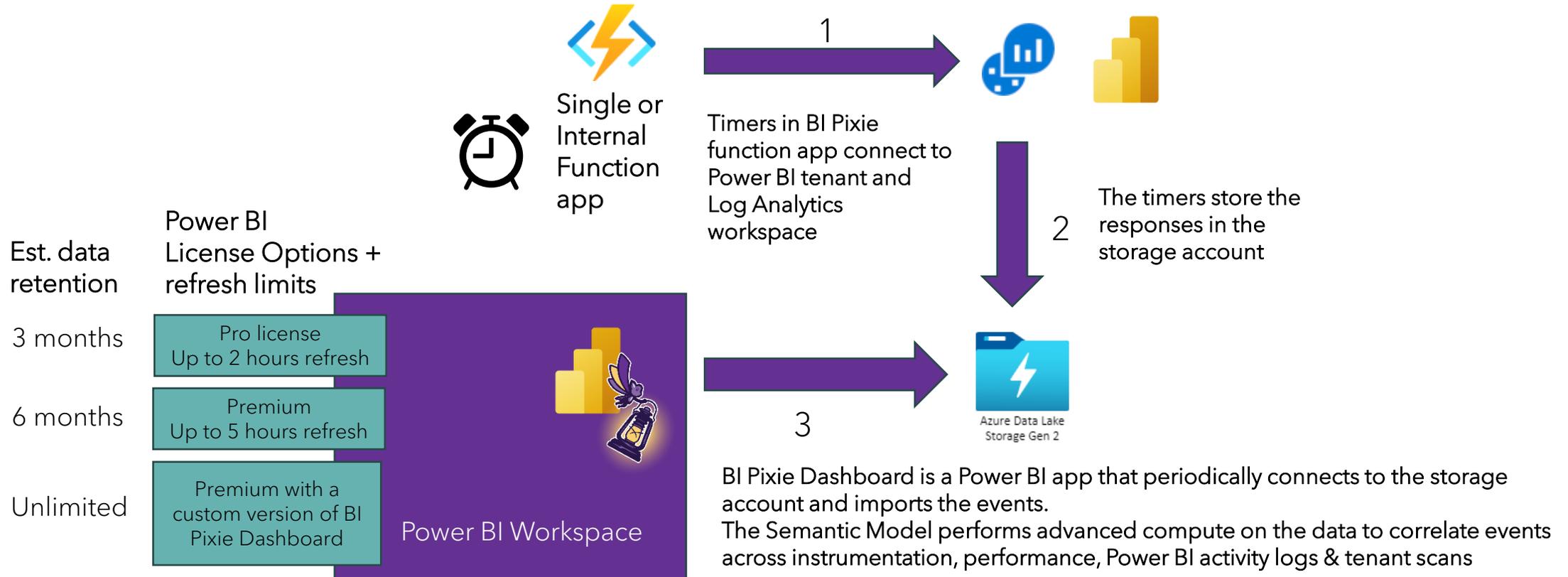


- All components can be installed and ready to be used in minutes.
- All data stay on your Azure account.



# BI PIXIE WORKFLOW – TIMERS AND SCHEDULE REFRESH

Your Azure environment



- All components can be installed and ready to be used in minutes.
- All data stay on your Azure account.

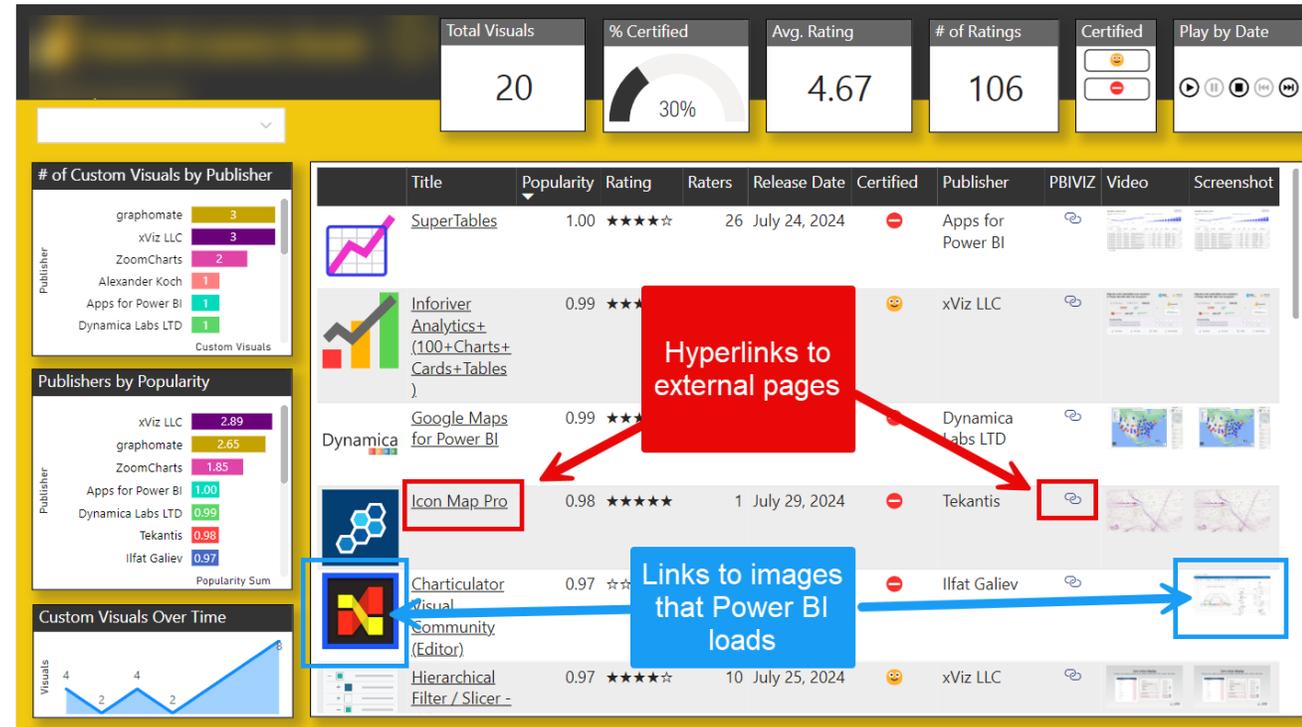


# EVENT TRACKING - BACKGROUND

## Background

- Power BI allows report developers to include static and dynamic URLs in visuals of the report pages
- These URLs served in two use cases:
  - Allowing end-user to click on an element and open a new page outside Power BI
  - Allowing end-user to view an external image within Power BI report
- The URLs in both cases can be created dynamically using measures in table visuals
- The creation of the URL is done at Power BI backend
- The web requests to fetch the images or access an external page is done on the client side

Real example: [here](#)





# EVENT TRACKING BY THE PIXIES

- BI Pixie instruments the reports by adding “Pixies” into the reports. The Pixies are built-in elements of Power BI: tables and measures that are configured to be hidden from end-users, but auditable by the BI team and IT.
- Each report page is added with one or more hidden single-cell table visual with a special measure that calculates a URL.
- The Pixie measure returns a URL to the Azure Function App that you have deployed:

```
https:// [your-function-app-name].azurewebsites.net/api/send_event?code=[your function app key]&[field1]=[value1]&[field2]=[value2]...
```

- The next slide describes the data that is generated in the URL query
- When the end-users load the report, their web client requests the URL above, and your Azure function serves back a pixel-sized image.



# THE DATA IN THE PIXIE MEASURES

## In any event:

- Report name
- Report page name
- Users (email address)
- CustomData (For embedded projects with app-owns-data)
- IP Address
- Power BI client platform (Web, Mobile, Desktop)
- User-Agent
- Event time
- Session duration

## In filter and slicer events:

- The columns used
- The min value and max value that the user sliced or filtered
- The distinct count of values users selected
- The visual name and visual type that was sliced

## In click events:

- The hyperlink that was clicked
- The bookmark name
- The visual name and visual type where the hyperlink was clicked (e.g. tables, matrices, buttons)

You can turn off any of the fields on all reports or on individual ones



# COLLECTED DATA DURING INSTRUMENTATION

For each instrumented report:

- Report name
- # of bookmarks
- Pages in the report

This data is sent as a Json file to the function app via POST request:

```
https:// [your-function-app-name].azurewebsites.net/api/send_event?code=[your function app key]
```

For each page in the report:

- Report page name
- Report page index (can be masked)
- # of visuals
- # of slicers
- # of table / matrix visuals
- # of columns in table/matrix visuals
- # of shapes/images/buttons
- true/false tooltip page
- true/false drill through page



## EVENT TRACKING BY THE PIXIES - HYPERLINKS

- To handle hyperlinks in Power BI, BI Pixie wraps up existing hyperlinks with a redirect logic.
- The Pixie wrapper measures are in the following format:

```
https:// [your-function-app-name].azurewebsites.net/api/send_event?code=[your function app key]&[field1]=[value1]&[field2]=[value2]...&ur=[original URL]
```

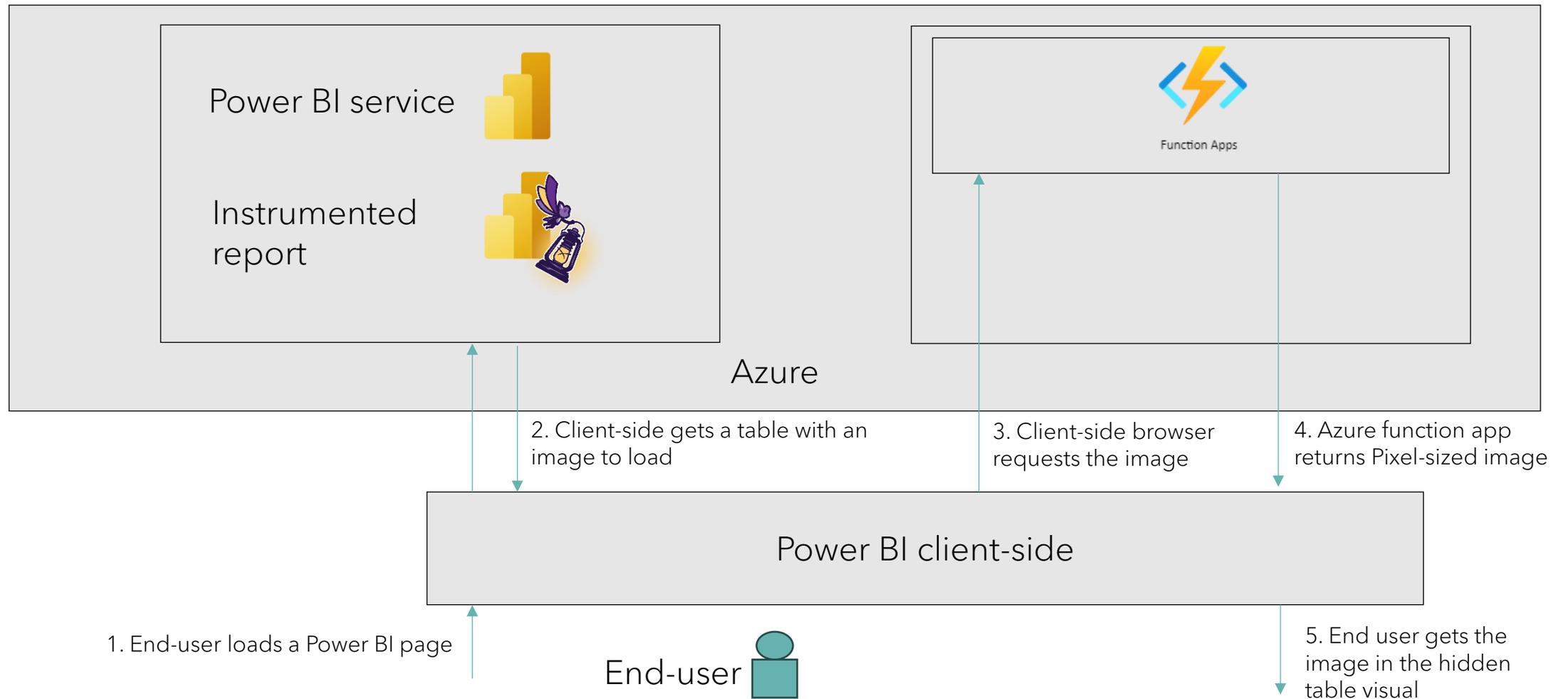
- When the end-users clicks on the hyperlinks, their web client requests the URL above, and your Azure function returns a redirection response:

```
HTTP/1.1 302 Found
```

```
Location: [original URL]
```

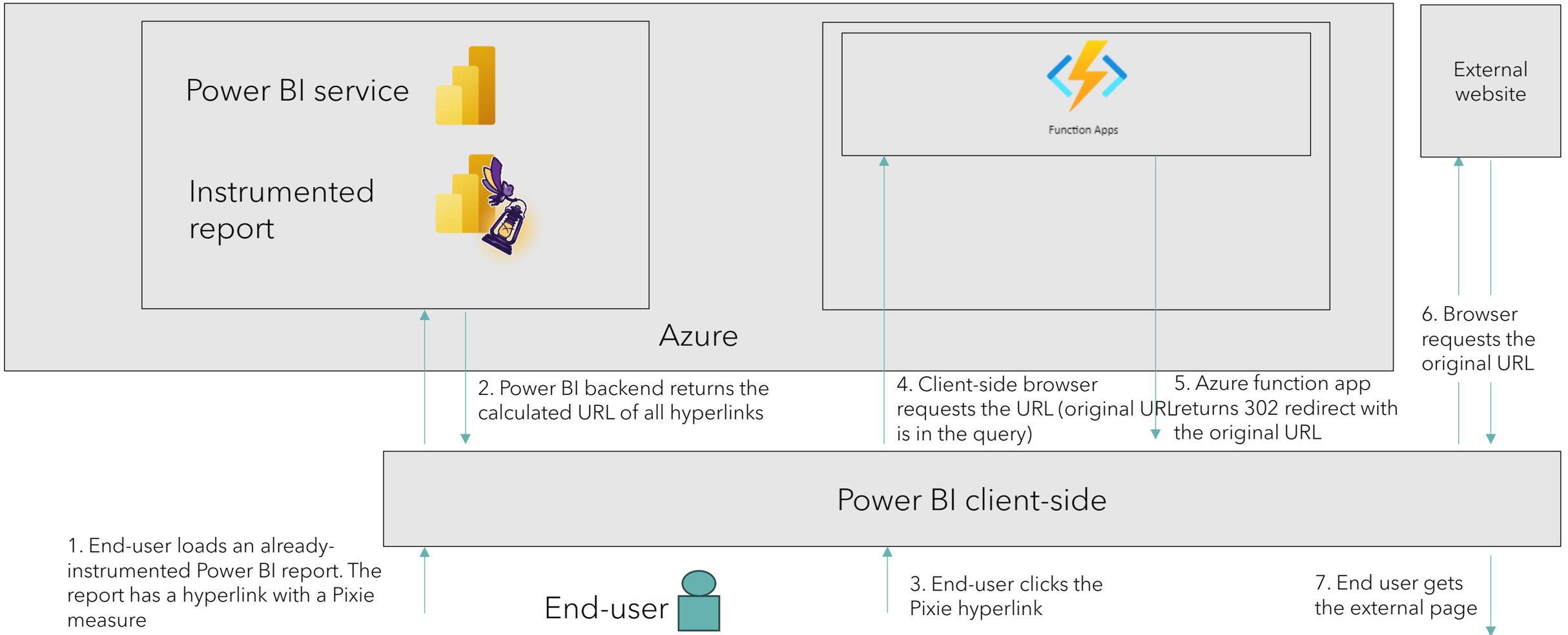


# TRACKING EVENTS (NOT LINK CLICKS)



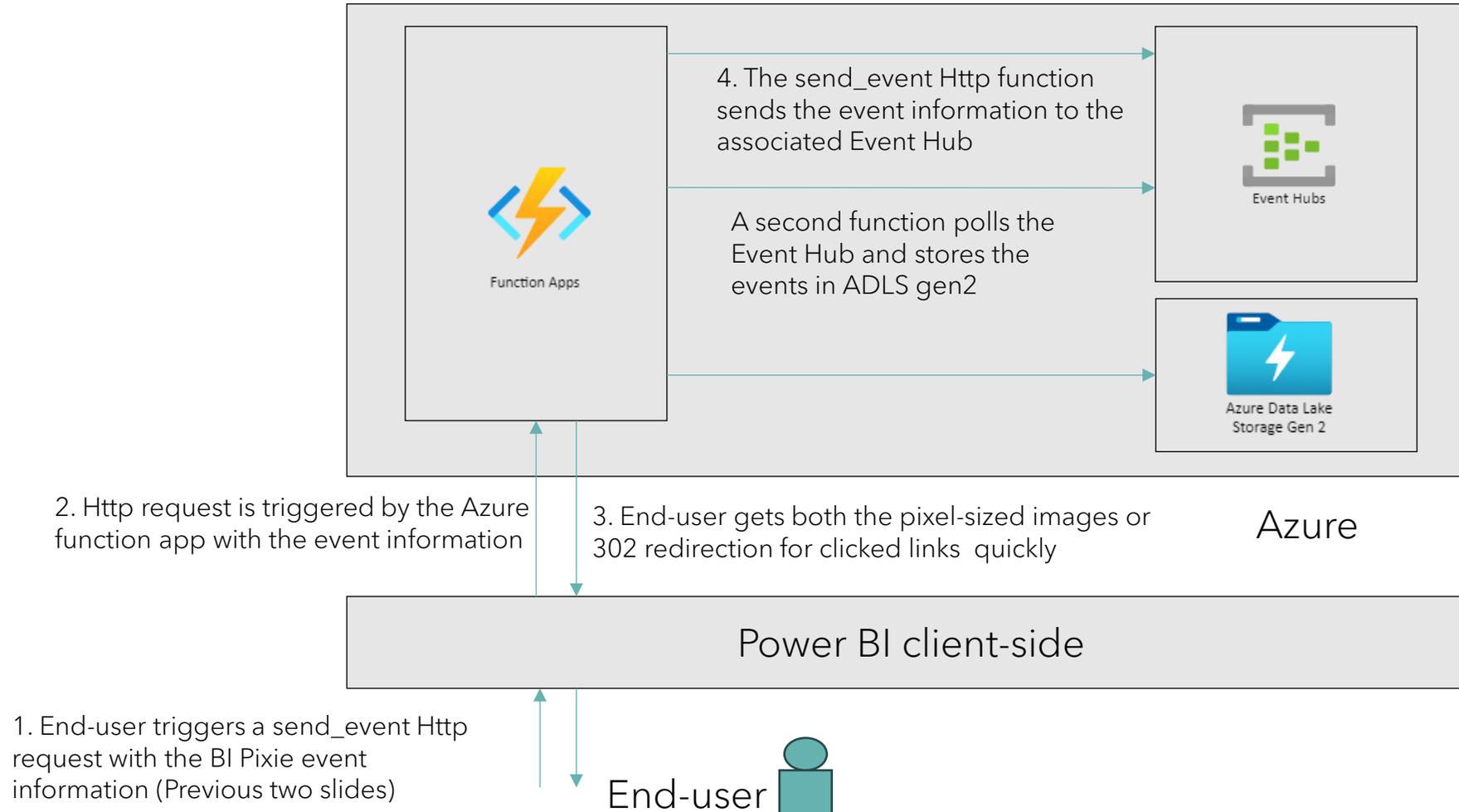


# TRACKING LINK CLICKS



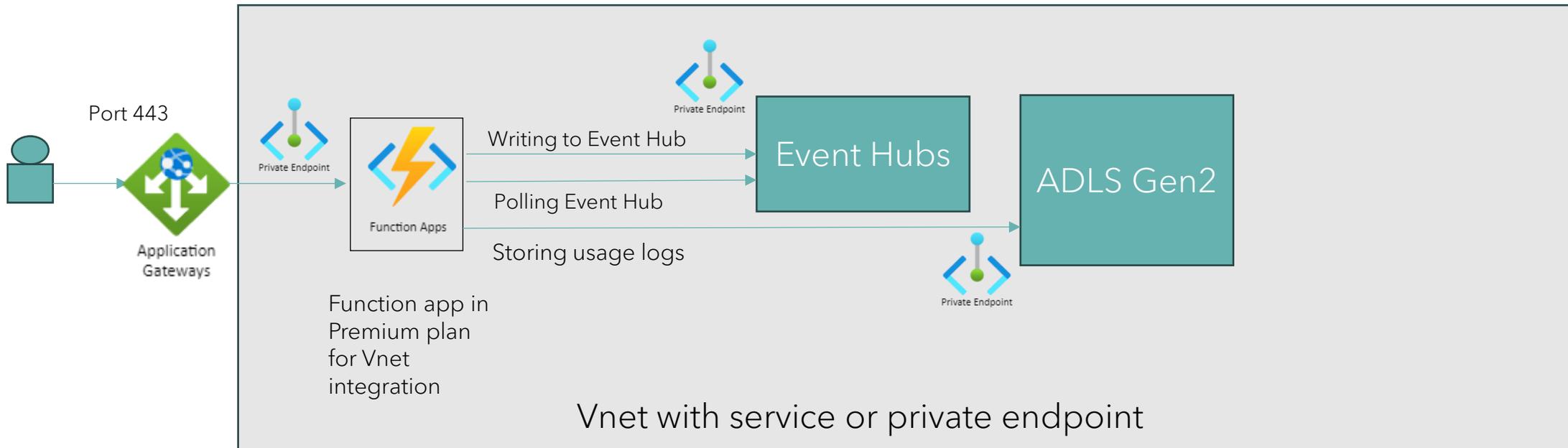


# FROM EVENT TO STORAGE



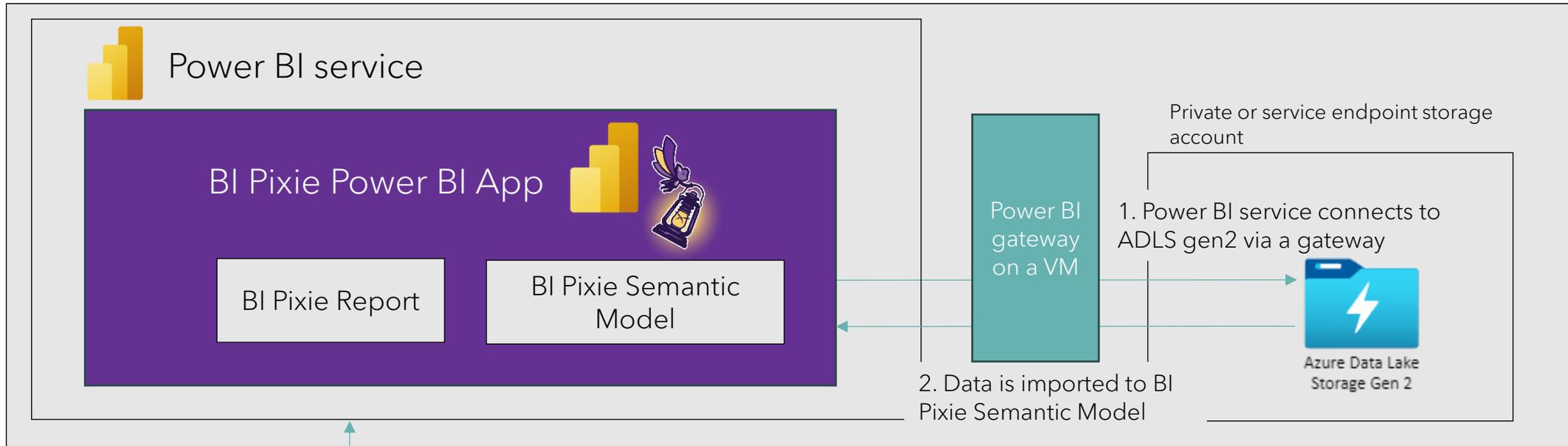


# VNET IN SINGLE-MODE ARCHITECTURE





# ANALYTICS USE-CASE WITH SECURED STORAGE



Azure

1. BI analysts open BI Pixie Report on Power BI to analyze the engagement of their audience

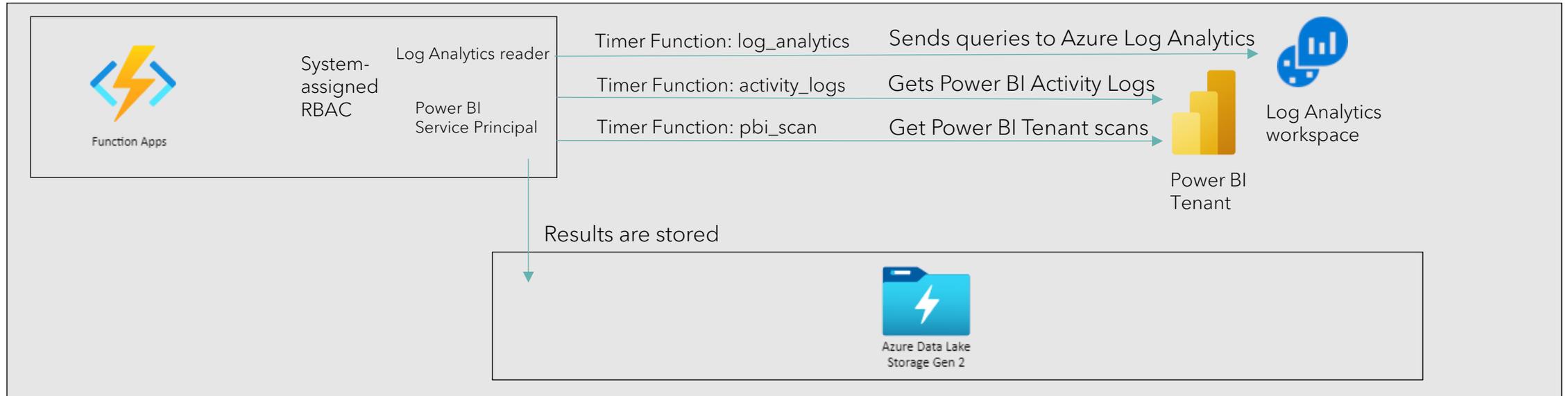
BI analyst





# OPTIONAL TIMERS

3 timers can be triggered by the Azure function app

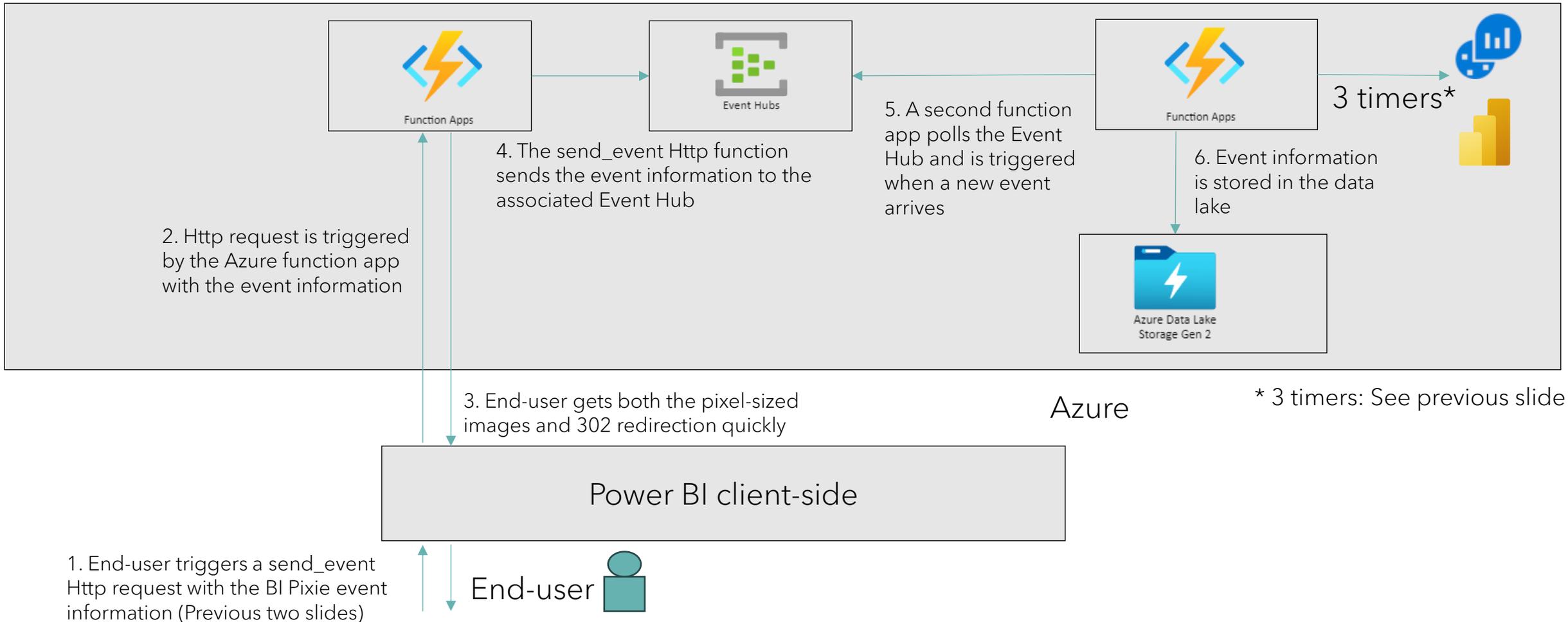


In the Function App in Single mode or the Internal Function App in Dual mode

Azure

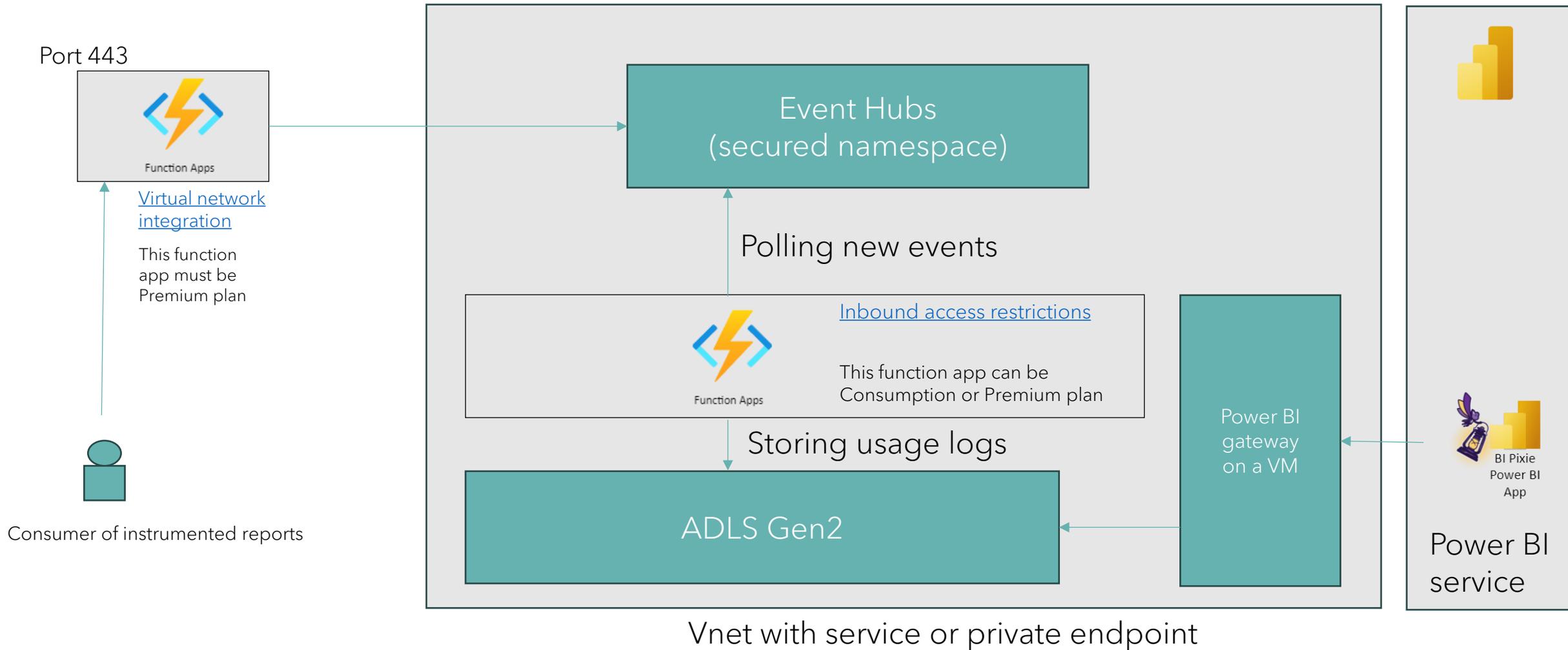


# FROM EVENT TO STORAGE – SINGLE MODE ARCHITECTURE





# VNET IN ADVANCED ARCHITECTURE (2 FUNCTION APPS)





## DATA AT REST (ADLS GEN2)

- Data is stored in your Azure Storage (Data Lake Storage Gen2 with Hierarchy Namespace)
- Azure Storage uses service-side encryption (SSE) to automatically encrypt your data when it is persisted to the cloud. Azure Storage encryption protects your data help you to meet your organizational security and compliance commitments.
- Data in Azure Storage is encrypted and decrypted transparently using **256-bit AES** encryption, one of the strongest block ciphers available, and is **FIPS 140-2 compliant**. Azure Storage encryption is similar to BitLocker encryption on Windows.
- Source: [here](#).



# DATA AT TRANSIT – AZURE APP SERVICE

BI Pixie is deployed as an Azure Function App and inherit the security features of Azure App Service. Your app allows TLS 1.2 by default, which is the recommended TLS level by industry standards, such as PCI DSS. You can enforce different TLS versions. Learn more [here](#)

In addition, Standard plan supports end-to-end TLS [here](#). Premium supports, minimum TLS cypher suite (TLS\_AES\_256\_GCM\_SHA384, TLS\_AES\_128\_GCM\_SHA256).

The screenshot shows the Azure Portal Configuration page for a Function App named 'bipixietestfunc1'. The 'General settings' tab is selected. The 'Minimum Inbound TLS Version' is set to 1.3. Other settings include Stack (Python), Python Version (Python 3.11), HTTP version (1.1), and HTTPS Only (On). The 'Configuration' option in the left-hand navigation menu is highlighted with a red box.



## ADDITIONAL SECURITY OPTIONS

- Azure Function App can be implemented using private endpoints (learn [here](#)).
- In such implementation, BI Pixie can work when end-users consumes the instrumented Power BI reports when they are connected VPN
- Custom domains and certificates: If you choose to use custom domain, you can secure it with TLS/SSL binding (learn [here](#)).