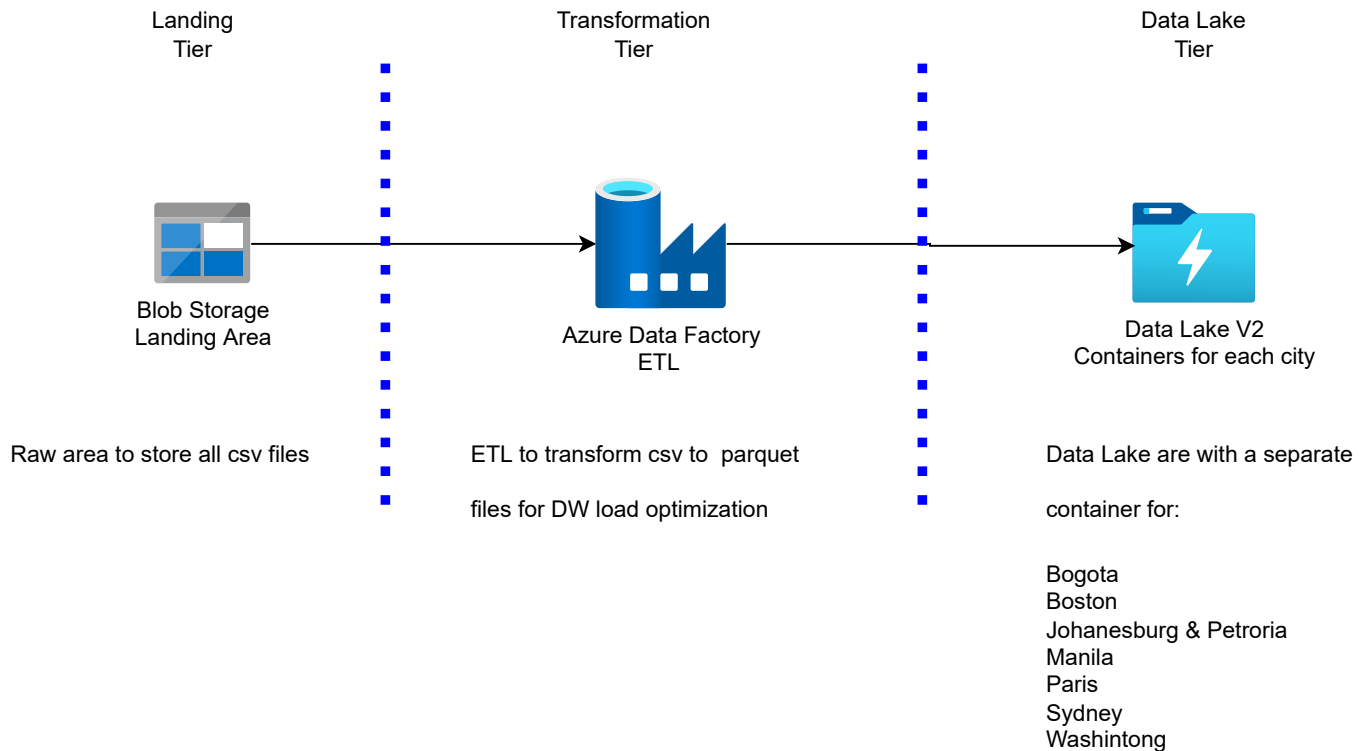


Travel Time Uber Movement General Architecture



We tackle the problem with the creation of 3 tiers 2 for storage and 1 for processing. The first tier is the landing tier where is going to be our staging storage to lad the CSV files to be consumed by ETL process that are going to transform them into .parquet files for better performance in the querying and ingestion into the DWH. The last tier is the datalake where we are going to use the hierarchical level to store the newly created parquet files

Security

Secure Transfer (Https)

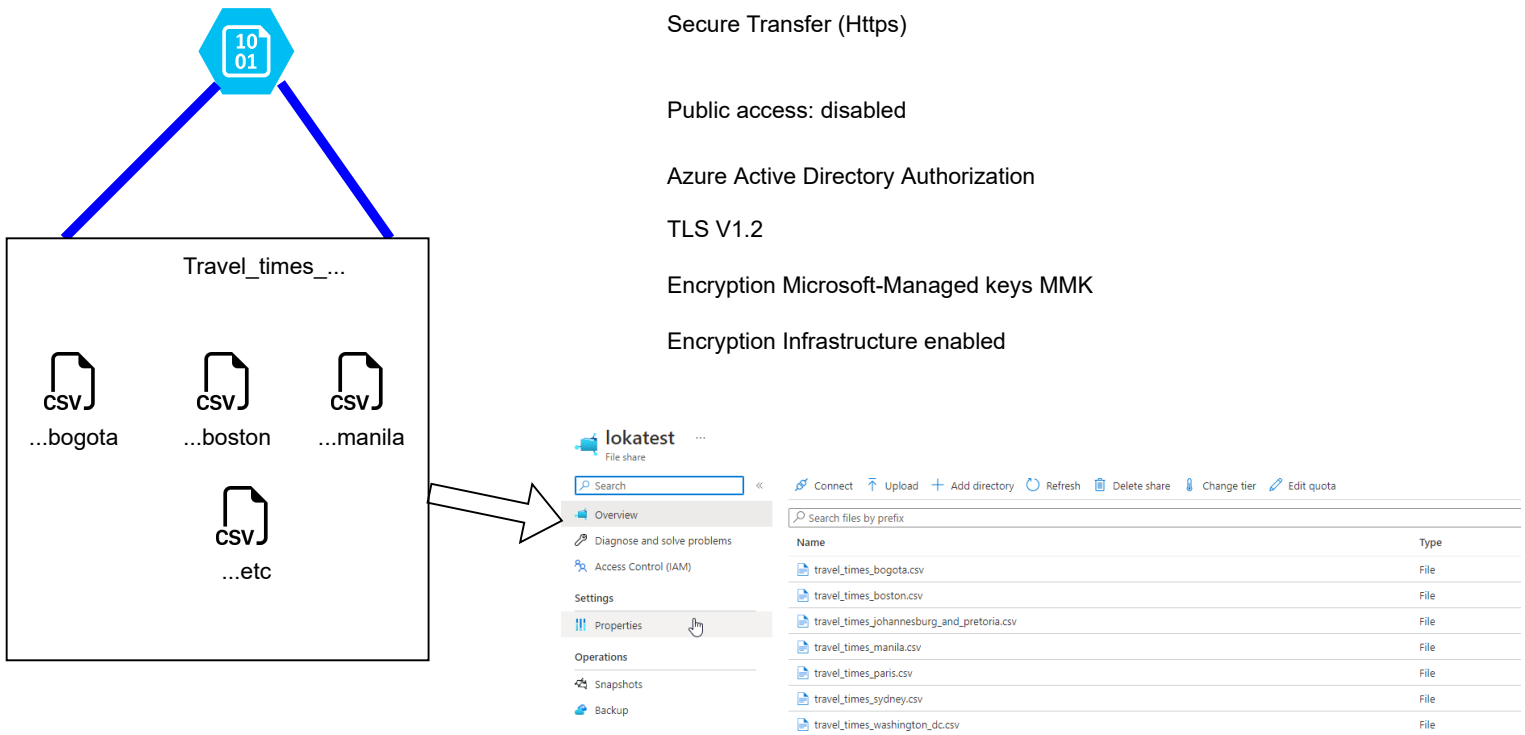
Public access: disabled

Azure Active Directory Authorization

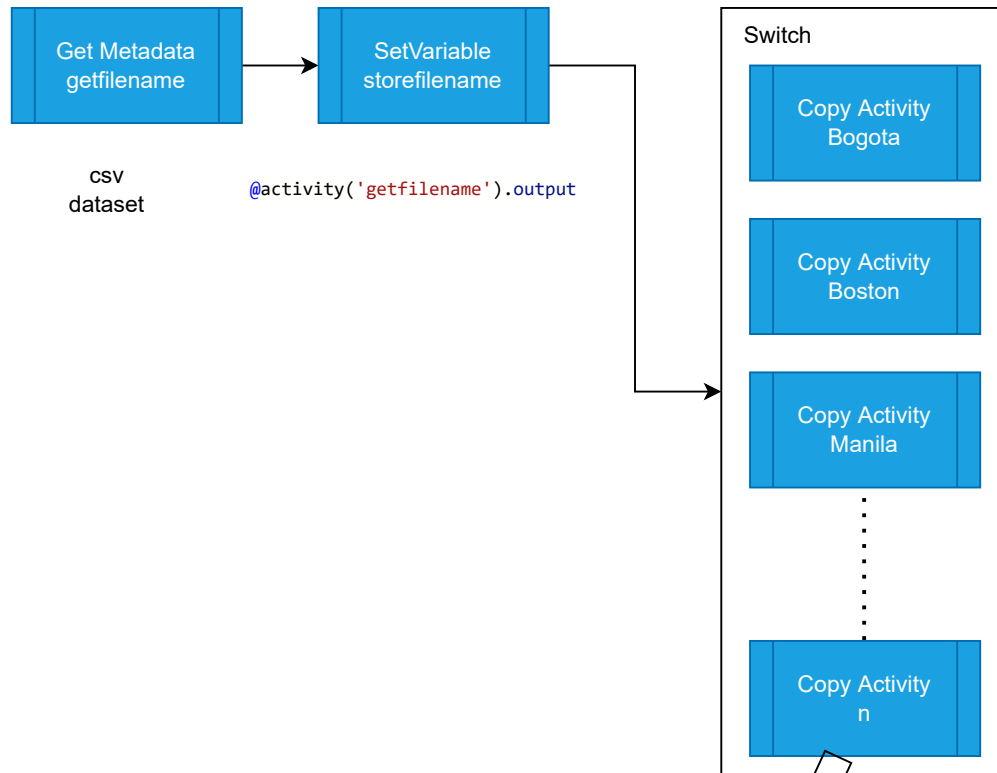
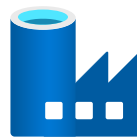
TLS V1.2

Encryption Microsoft-Managed keys MMK

Encryption Infrastructure enabled



The purpose of the landing zone is to leverage the basic blob storage service provided by Azure in order to stage the raw csv file with the uber movements data. No hierarchy is needed here as per this is going to be populated by a simple transfer service in order to be consumed by ADF (see previous page)



Source dataset *
Source dataset: [Open](#) [+ New](#) [Preview data](#) [Learn more](#)

File path type
☐ File path in dataset ☐ Prefix ☒ Wildcard file path ☐ List of files

Wildcard paths
lokatest / /

Start time (UTC) End time (UTC)



blob

Sink dataset *
Sink dataset: [Open](#) [+ New](#) [Learn more](#)

Copy behavior



data lake

```
@concat('/',string(formatDateTime(utcNow(),'yyyy')), '/',string(formatDateTime(utcNow(),'yyyy')), '/',string(formatDateTime(utcNow(),'MM')),  
'/',string(formatDateTime(utcNow(),'dd')), '/Bogota_', formatDateTime(utcNow(),'yyyy-dd-MM'), '. parquet ')
```

ETL Process

Leveraging Azure Data Factory 2 data sets are required. 1 for the source (blob shares) and a second for the destination (data lake). ETL is going to read the file, get the name, validate the last portion of the string to check what city belongs to, with this name it is going to process the copy activity assigning a hierarchy path using the city name + date + a specific file name. In this process ADF converts the file to parquet using a data set for this purpose



bogota
Container

Search [] Upload + Add Directory Refresh [] Rename [] Delete [] Change tier [] Acquire lease [] Break lease

Authentication method: Access key ([Switch to Azure AD User Account](#))
Location: bogota / 2022 / 11 / 22

Search blobs by prefix (case-sensitive) ☐ Show deleted objects

Name	Modified	Access tier	Archive status	Blob type	Size	Lease state
[] [] { }						***
[] Bogota_2022-11-22.parquet	11/22/2022,6:23:34 PM	HOT (preferred)		Block blob	617.08 KB	Available ***

[Home](#)
[About](#)
[Contact](#)
[Privacy Policy](#)
[Terms of Service](#)
[Sitemap](#)

Private endpoints, vpn, and IP addresses

MS Network Routing

Infrastructure Encryption Enabled

Encryption type : Microsoft - managed keys

MS Network Routing

Infrastructure Encryption Enabled

Encryption type : Microsoft - managed keys

Files are transformed from source shares from CSV to PARQUET. The reason for this is to leverage the advantage of the columnar stored format for speeding analytics queries or for data ingestion to the DWH (in this exercise we are assuming we are going to ingest to AZ synapse analytics that as well is a columnar storage). We can do query directly via any BI tool for example Power BI with the parquet connector.