Azure Storage Part II

CSCI E-94

Fundamentals of Cloud Computing - Azure

Joseph Ficara

Agenda

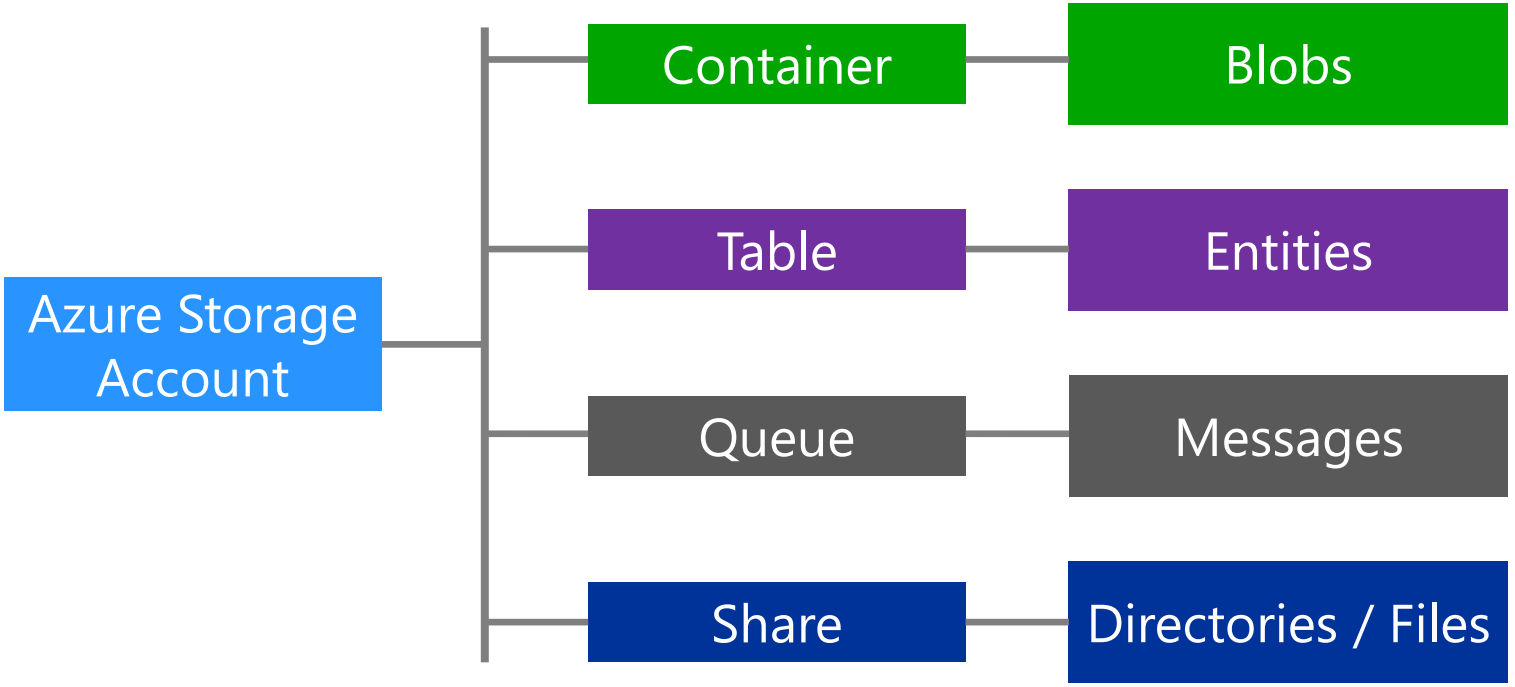
- Overview
  - Refresher…
  - Tables, Queues, Files
- Azure Table Storage
  - Implementation
- Azure Queues
  - Implementation
- Azure File Basics

Overview
Refresher …

- "*Azure Storage*" has 4 storage facilities
  - Blobs
  - Queues
  - Tables
  - Files

Overview
Refresher …

- Azure Storage resources relationships



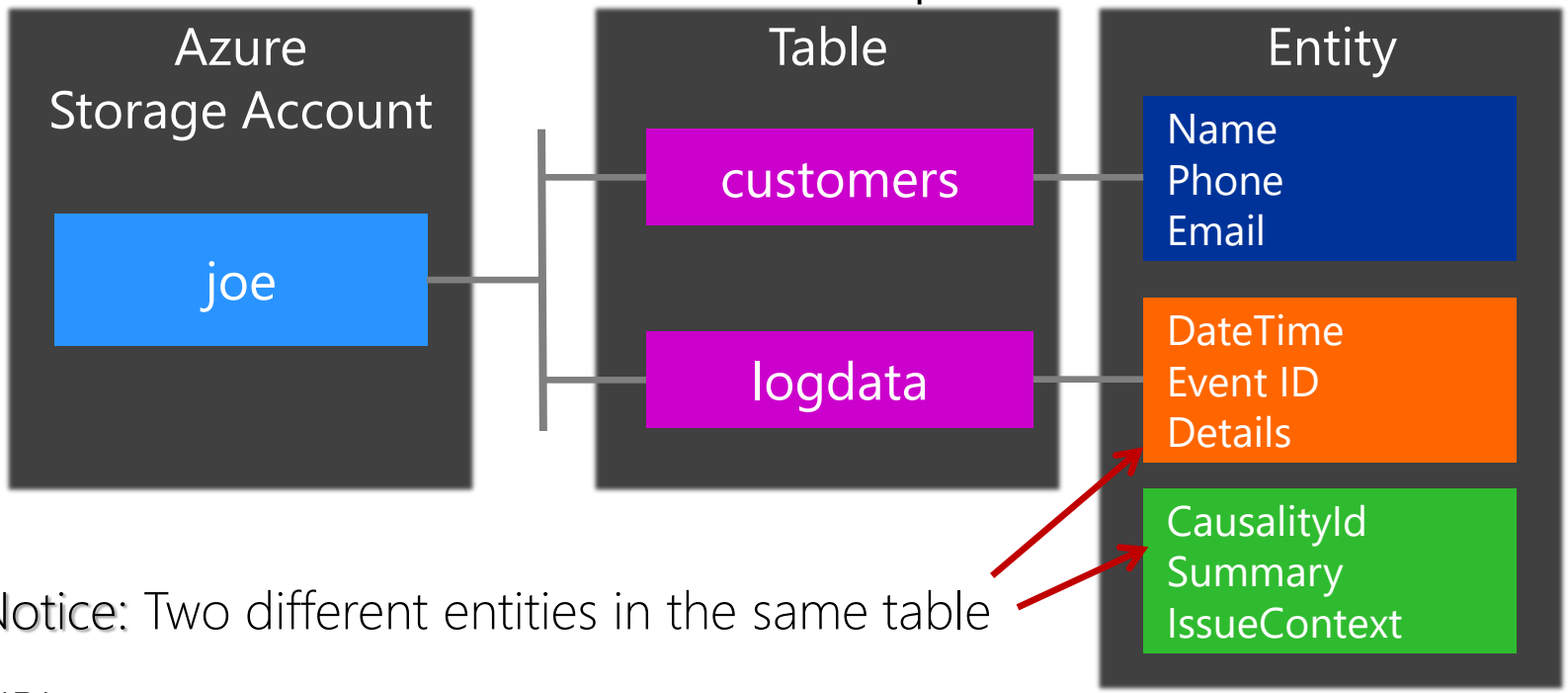
See: Azure Storage Concepts: http://bit.ly/3twKleu

Overview
Azure Table Storage

- Microsoft's NoSQL key/attribute store
  - Schema-less
- Typically, lower cost than SQL
- Heterogeneous table entities
  - Each "row" or Entity can be different
    - Different "columns" Properties
- Entity limit is bounded by storage account
  - No fixed entity limit per se

Overview
Azure Table Storage

- Table Service Concepts



Notice: Two different entities in the same table

URL to access queue:
http://<storage account>.table.core.windows.net/<table>

Adapted from: What is Azure Table Storage

Overview
Azure Table Storage

- Common Uses
  - Storing large amounts of structured data
    - Serving web scale applications
    - Data that does not require complex
      - Joins, foreign keys or stored procedures
    - Denormalized for fast access
    - Quick access via “clustered” index
    - Performing post process analytics (Big Data)
      - Heat maps
        - What’s the most common log error by time of day
        - Customer ordering patterns per region

Overview
Azure Table Storage

- Rich client libraries
  - .NET
  - Java
  - C++
  - Ruby
  - Node.js
- Accessible via REST for everyone else
  - PHP & Python
- Also supports subset of OData Protocol

Overview
Azure Table Storage

- Supported data types
  - Binary
    - Array of bytes up to 64 KB
  - Bool
    - A Boolean value
  - DateTime
    - 64-Bit UTC Time 1/1/1601 to 12/31/9999
  - Double
    - 64-Bit floating point value

Overview
Azure Table Storage

- Supported data types …
  - GUID
    - 128-bit globally unique identifier
  - Int
    - 32-bit integer
  - Int64
    - 64-bit integer
  - String
    - UTF-16 encoded value up to 64 KB

Overview

Azure Table Storage

- Comparison to Azure SQL:

Criteria	Azure Table Storage	Azure SQL Database
Data Relationships	No	Yes
Server-side processing	No No foreign keys, stored procs, triggers, etc…	Yes
Transaction support	Limited Same table & partition, Up to 100 operations, Optimistic concurrency	Yes
Geo-Replication	Yes	Yes
Table schema	Relaxed Heterogeneous entitles	Managed
Similarity to on prem DB	No	Yes
Scale-out	Automatic	Manual
Data types	Simple	Simple, Complex, User Defined

Overview
Azure Table Storage

- Costs as of 02/25/2025

  Many costs are region specific
  See Azure Calculator: http://bit.ly/1cMjBOY

  - Storage
    - $0.045 per gigabyte stored per month
  - Operations and Data Transfer

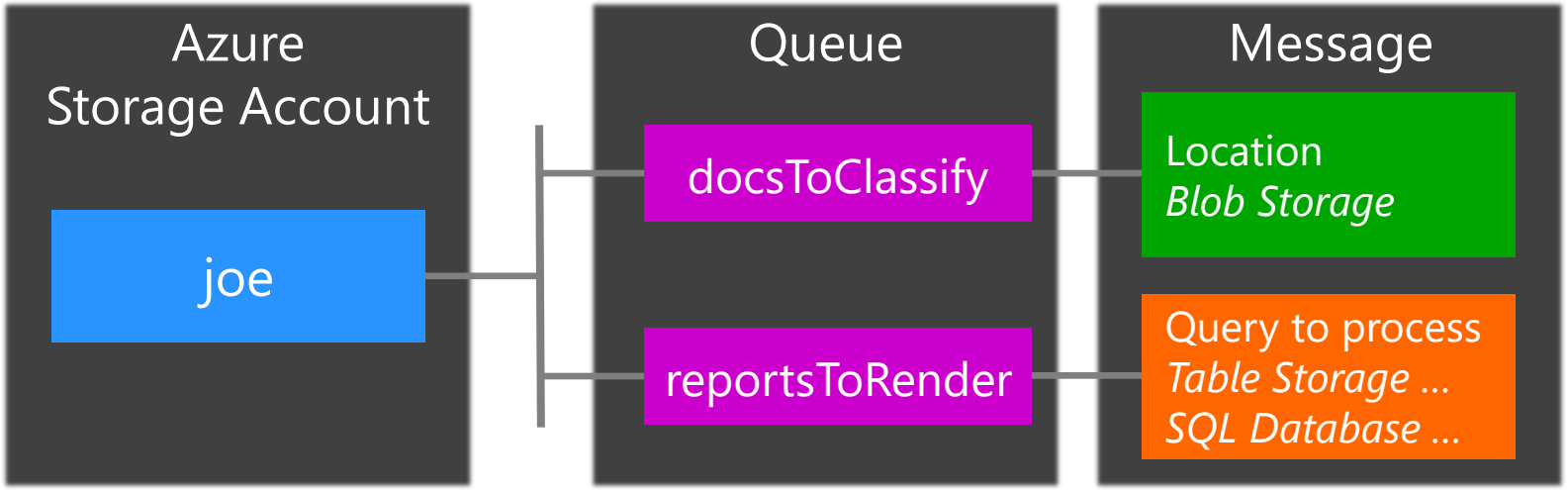
    List & Create, Read, Other, Data Retrieval Data Write – See Azure Calculator
    - Total Cost: $0.04 Per Month for 1 million transactions
  - Geo-replication redundancy (GRS)
    - $0.06 – per gigabyte stored per month

Overview
Azure Table Storage

Overview

Azure Queues

- Queue Service Concepts



URL to access queue:
http://<storage account>.queue.core.windows.net/<queue>

Adapted from: Introduction to Azure Queue Storage

Overview
Azure Queues

- Common use cases
  - Producer consumer pattern
    - A backlog of work to process asynchronously
  - Passing messages
    - Azure Web Role to Azure Worker Role
  - Connecting dependencies
    - Job Type 1
      - Finishes its part submits result to queue
    - Job Type 2
      - Processes work submitted by Job Type 1

Overview

Azure Queues

- Core Features
  - Enqueue / Dequeue
    - Standard for adding and retrieving message
  - Message Peek
    - Look at message in front of queue
      - Without dequeuing
  - Changing message contents
    - Of a message in the queue

Overview

Azure Queues

- Core Features
  - Message Lifetime
    - How long the message lives in the queue
  - Message visibility delay
    - Delay before a message is visible in the queue
      - Scaling / Scheduling for processing in the future
  - Dequeue Count
    - Useful for implementing poison message support
  - Queue length
    - Useful for auto scaling

Overview
Azure Queues

- Core Features
  - Message lease time
    - Dequeued messages
      - Automatically become invisible
    - Visibility automatically returned
      - If not deleted within lease time specified
        - Lease time is called invisibility time in docs
    - Prevents a worker process from killing messages
      - Assumption:
        - If a worker fails to call delete assume the worker failed
      - Possible to extend invisibility time
        - Update message in the queue

Overview

Azure Queues

- Extended features from other SDKs
  - Poison Messages (WebJobs SDK)

    WebJobs SDK How to use the WebJobs SDK
    - Based on configurable retries
    - Poison queue is named
      - {originalqueuename}-poison

Overview
Azure Queues

- Rich client libraries
  - .NET
  - Java
  - C++
  - Ruby
  - Node.js
- Accessible via REST for everyone else
  - PHP
  - Python

Overview

Azure Queues

- Azure Queue Costs
  - Class 1 Operations
    - CreateQueue
    - ListQueues
    - PutMessage
    - SetQueueMetadata
    - UpdateMessage
  - Class 2 Operations
    - GetMessage
    - GetMessages
    - GetQueueMetadata
    - GetQueueServiceProperties
    - GetQueueAcl
    - PeekMessage
    - PeekMessages
    - GetMessageRead
    - GetMessagesRead.

Overview

Azure Queues – Costs 2/20/2025

- 1000 GB Locally Redundant $45.00 USD
  - 100 Million Storage **Class 1** operations: **$40.00**
  - 100 Million Storage **Class 2** operations: **$40.00**
- 1000 GB Geo Redundant $60.00 USD
  - 100 Million Storage **Class 1** operations: **$80.00**
  - 100 Million Storage **Class 2** operations: **$40.00**
- Queue Capacity per GB:
  - **LRS** $0.0450, **GRS** $0.0589
- Azure Calculator: http://bit.ly/1cMjBOY
  - Costs very by region

Overview

Azure Queues

- Limits http://bit.ly/2V9vQi6
  - Max size of
    - single queue: **500 TiB**
    - message in a queue **64 KiB**
  - Max number of
    - stored access polices per queue **5** http://bit.ly/2wwpXRF
  - Maximum request rate per storage account
    - **20,000** message per seconds **1 KiB** msg size
  - Target throughput single queue **1 KiB** msg
    - Up to **2000** messages per seconds

Overview

Azure Queues

Overview
Azure File Storage

- What is Azure File Storage
  FAQs: http://bit.ly/2uYp2cA
  - Shared storage using SMB 3.x protocol
  - Network File System (NFS v4.1)
  - A way to expose file shares from Azure
    - Access data using standard file I/O APIs
      - Accessible to VMs running in Azure
  - You can Map/Mount
    - Windows, Linux, MacOs

Overview
Azure File Storage

- Map Drives / Mounting
  - Supports Windows, Linux, and Mac OS
    - Port: 445 needs to be open!
      - Check with PortQry: http://bit.ly/2EU5slq
      - Use AzFileDiagnostics: http://bit.ly/3pozZdn
    - See: http://bit.ly/2sXa30K
    - Trouble shooting see:
      - http://bit.ly/2BRRSwh
- Can also access via
    - File storage REST API
    - PowerShell

Overview
Azure File Storage

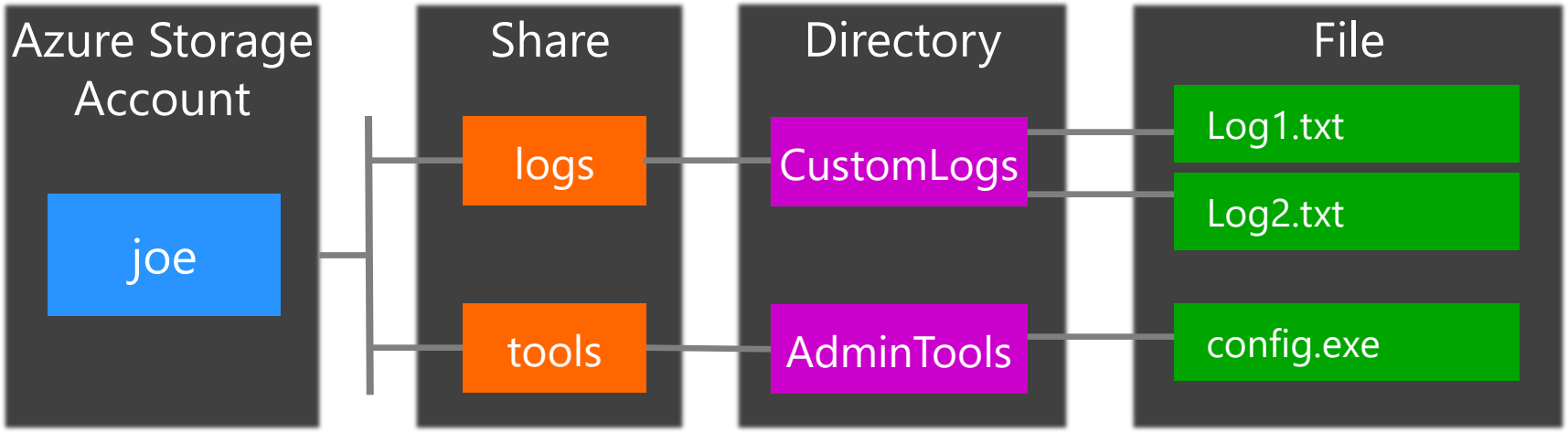
- Remote access from PowerShell?
  - Yes, you can access from PowerShell
    - From an on-premises server
  - Need Azure PowerShell version 1.0 or later
    - See: Install Azure PowerShell with PowerShellGet

Overview
Azure File Storage

- Common use cases
  - Migrating an on-premise solution to Azure
  - Storing shared application settings
    - Configuration files
  - Storing diagnostic data
    - Logs, metrics, crash dumps in shared location
  - Storing tools and utilities for Administrating
    - Azure virtual machines
    - Cloud Services

Overview
Azure File Storage

- File storage concepts



- Azure File scale targets
  - Unlimited number of shares, and files (optional directory hierarchies)
    - Files up to 4 TiB in size each up to 100TiB Per share as of 02/25/2025
      - See: http://bit.ly/3diZBWI

URL to access:
http://<storage account>.file.core.windows.net/<share>/<directory/directories>/<file>
Adapted from: Introduction to Azure Files

Overview
Azure File Storage

- What about costs?
  - Low cost - East US of 2/25/2025

    Performance Tier: Hot (Pay as you go)
    - 1000 GB Locally Redundant
      - Data at-rest $28.70 USD
      - Metadata at-rest $29.70 USD
    - 1000 GB Geo Redundant
      - Data at-rest $63.2 USD
      - Metadata at-rest $65.5 USD
    - Sync servers
      - $5.00 USD per additional sync server
      - One sync server is free per storage sync service
    - See Azure Files Pricing: http://bit.ly/2EUrrsc
      - Many more details …

Overview

Azure File Storage

Azure Table Storage
Implementation – New SDK

- There is a new SDK
  - **Azure.Data.Tables** nuget package
  - Be sure to also install
    - **Microsoft.Extensions.Azure** nuget package
  - Steps are similar but streamlined
- You need a class that defines the schema
  - Derive from ITableEntity
  - Includes the following properties
    - PartitionKey, RowKey, Timestamp and ETag

Azure Table Storage
Implementation – New SDK

- Access to table functions provided by
  - **TableServiceClient**
    - Table level operations
      - Create table, Delete Table GetStatistics
  - **TableClient**
    - Allows interaction with a table
      - In Cosmos or AzureStorage
    - Query, GetEntity, UpdateEntity, DeleteEntity

Azure Table Storage
Implementation

- Requires two specific properties
  - RowKey
    - Must be unique
  - PartitionKey
    - Doesn't have to be unique
    - Defines a partition boundary
      - Keep like data in same partition

Azure Table Storage
Implementation

- 5 Steps for adding an entity
  - 1. Add a TableServiceClient
    - With the connection string in program.cs
      - Dependency injection support
  - 2. Create / Attach to the table
    - In controller's constructor get injected TableServiceClient
    - Using the TableServiceClient
      - Attach to the table
      - Use the TableClient to
        - Create the table if it doesn't exist
          - CreateIfNotExistAsync

Azure Table Storage
Implementation – New SDK

- To support dependency injection
  - Use **AddTableServiceClient**
    - In **Program.cs**
  - Provide the connection string
    - During setup

Azure Table Storage

Register table service client

```
// Register the table service client
builder.Services.AddAzureClients(c =>
{
    c.AddTableServiceClient(
        builder.Configuration.GetConnectionString(
        TableConstants.TABLE_CONNECTION_STRING_NAME));
     }
);
```

Azure Table Storage
Implementation

- 5 Steps for adding an entity
  - 3. Set the RowKey and PartitionKey
    - The RowKey needs to be unique
    - The PartitionKey defines the scaling boundary
      - Azure will keep data in a partition together
        - Use the partition key to control how Azure will split tables
  - 4. Use the TableClient
    - To add the entity
      - Use the AddEntityAsync method
  - 5. Check the result for status of the operation
    - result.IsError

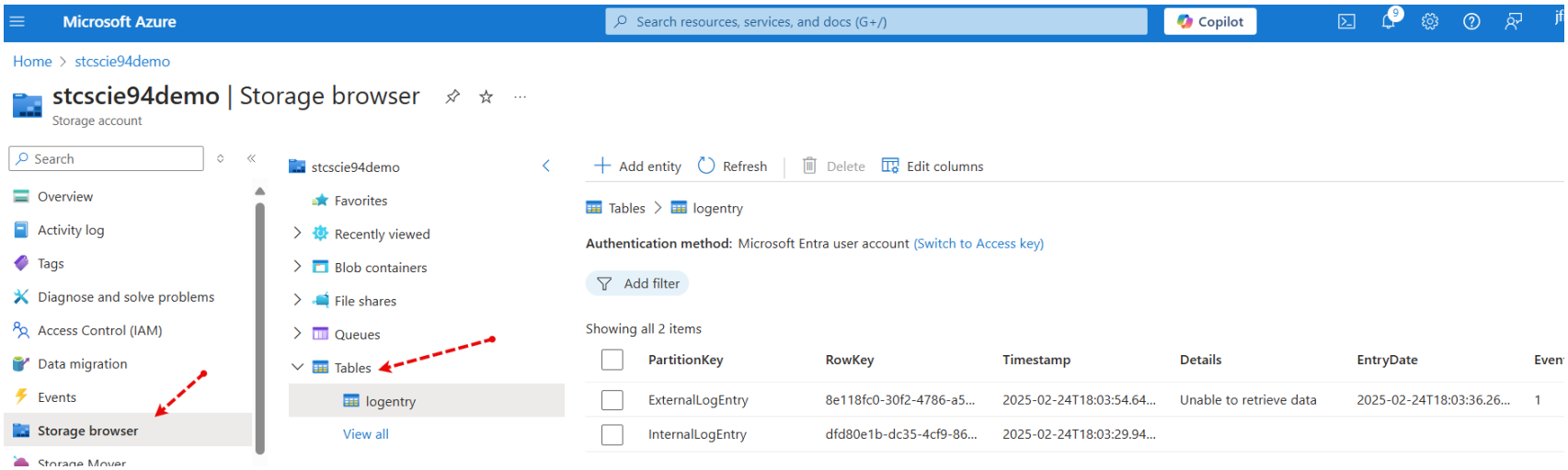
Azure Table Storage
Implementation – New SDK

- To retrieve an entity
  - Get a **TableClient** from
    - **TableServiceClient**
  - Call **GetEntity**
  - Check response status
    - Return result if success
  - Catch **RequestFailedException**
    - Return status code or translate if necessary

Azure Table Storage

Azure Portal - Storage Browser Tables

- The azure portal has a Tables browser
  - You need to have
    - **Storage Table Data Contributor**
    - See: Assign an Azure role for access to table data



Demo

Azure Table Storage

Implementation

`AzureTableDemo2Solution`

`Azure Portal Storage Browser : Tables`

Azure Queues
Implementation – New SDK

- There is a new SDK
  - **Azure.Storage.Queues** nuget package
  - Be sure to also install
    - **Azure.Storage.Common** nuget package
- Steps
  - Create a queue client
  - Perform the desired operation
    - SendMessageAsync
    - RetrieveMessageAsync
    - DeleteMessageAsync

Azure Queues
Implementation

- Adding Queue Message requires 4 steps
  - 1. Create a QueueClient
    - Using the connection string
  - 2. Create / Attach to the queue
    - Using the QueueClient
    - Create the queue if it doesn't exist
      - CreateIfNotExists

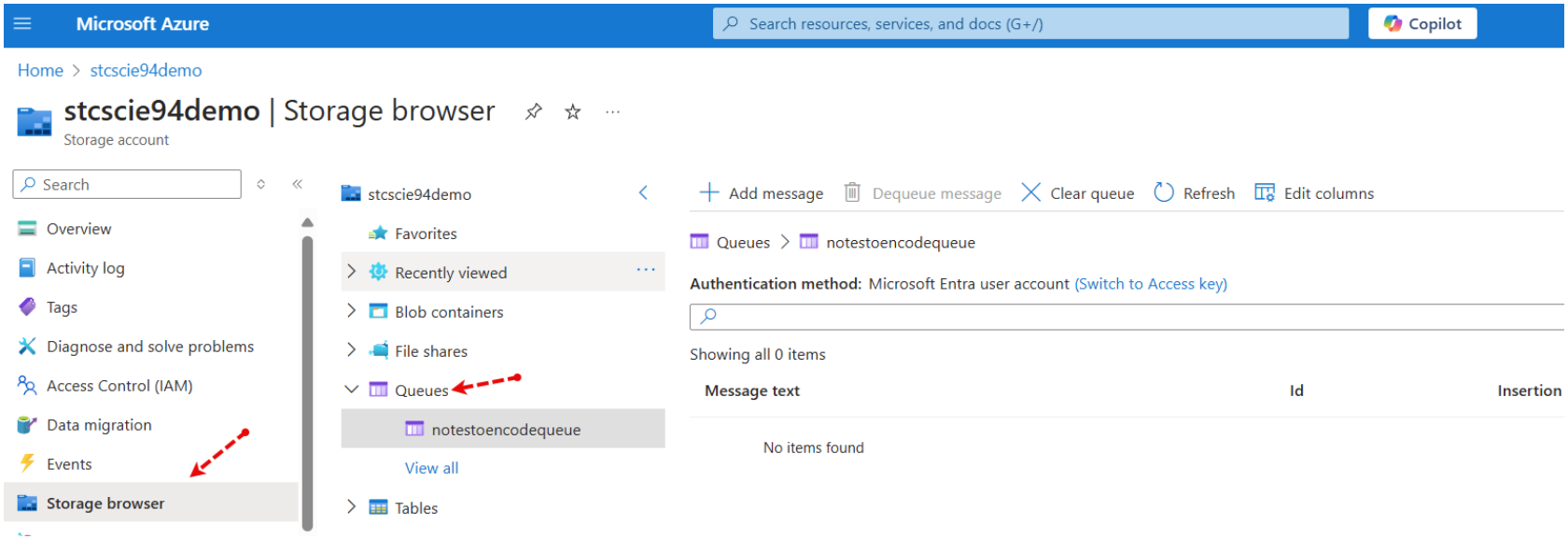
Azure Queues
Implementation

- Adding Queue Message requires 4 steps…
  - 4. Send a queue message
    - Specify time to live
      - Using a TimeSpan
  - 5. Verify response
    - Using the returned Response<SendReceipt>
      - Status property,
        - Checking for a StatusCode.Status201Created
  - Done!

Azure Queues

Azure Portal - Storage Browser Queues

- The azure portal has a Queue browser
  - You need to have
    - **Storage Queue Data Contributor**
    - See: Assign an Azure role for access to queue data



Demo

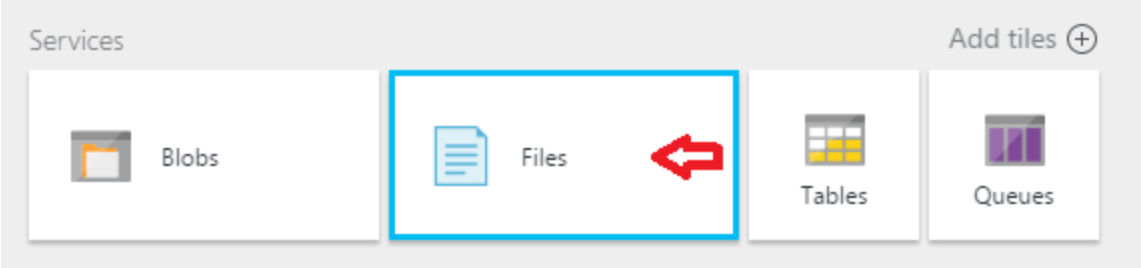
Azure Queues

Implementation

`AzureQueueDemo2Solution`

Azure File Storage

- Then create the storage account
  - Check dashboard for the newly created account



  - To create a share/directory/file via PowerShell
    - See:
      - Quickstart for creating and using Azure file shares

Azure File Storage

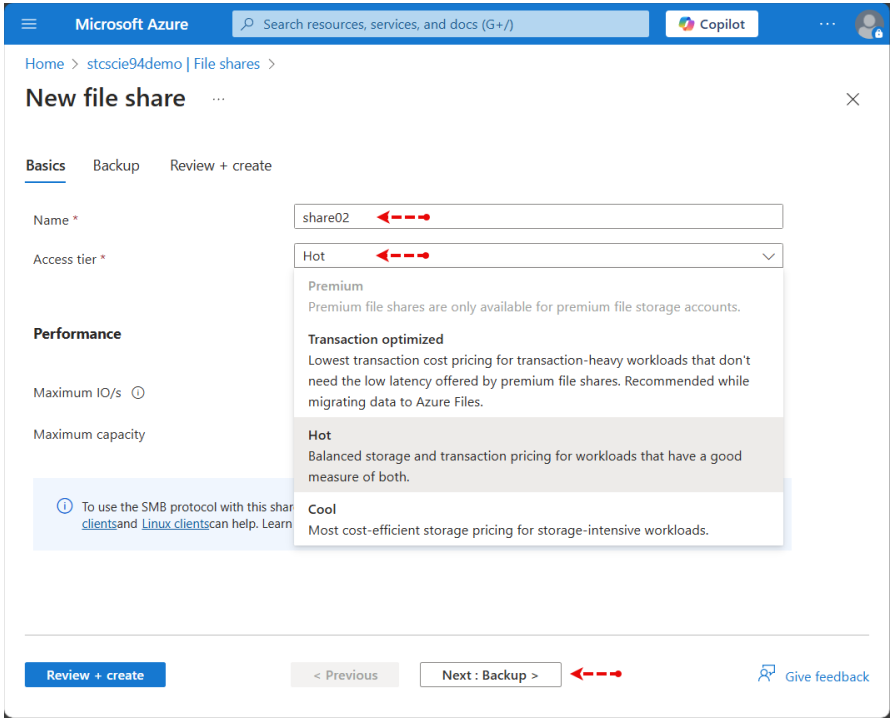
- Several ways to interact with File Storage
  - Map a drive letter from a
    - VM running in Azure
    - On premise machine Windows 8 or greater
      - Supporting SMB 3.0
      - Port 445 (TCP Outbound) needs to be opened
        - Some internet providers block port 445
  - PowerShell: See: http://bit.ly/2sUVP03
  - REST Interfaces: See: http://bit.ly/1pBCxGR
  - Azure Dashboard

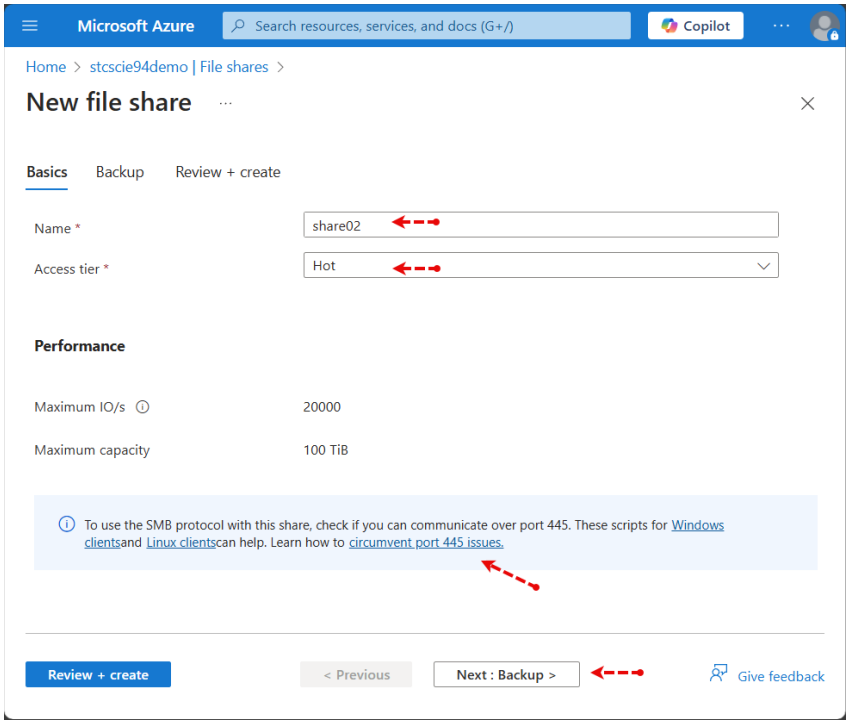
Azure File Storage

- Setting things up requires 4 basic items
  - 1. Storage Account
    - To create your share in
  - 2. Share
    - Is the mount point
    - Allows specification of a capacity quota
  - 3. Directory
    - Container of files, supports nested directories
  - 4. Files
    - The content

Azure File Storage

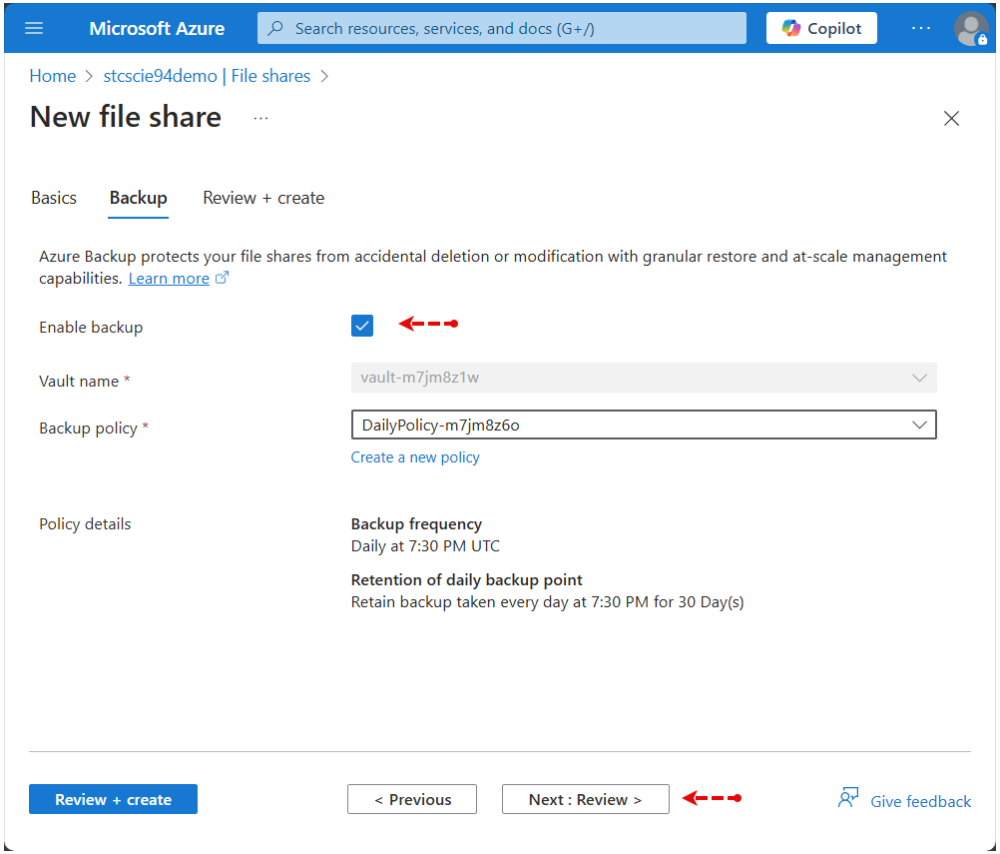
- Create a share





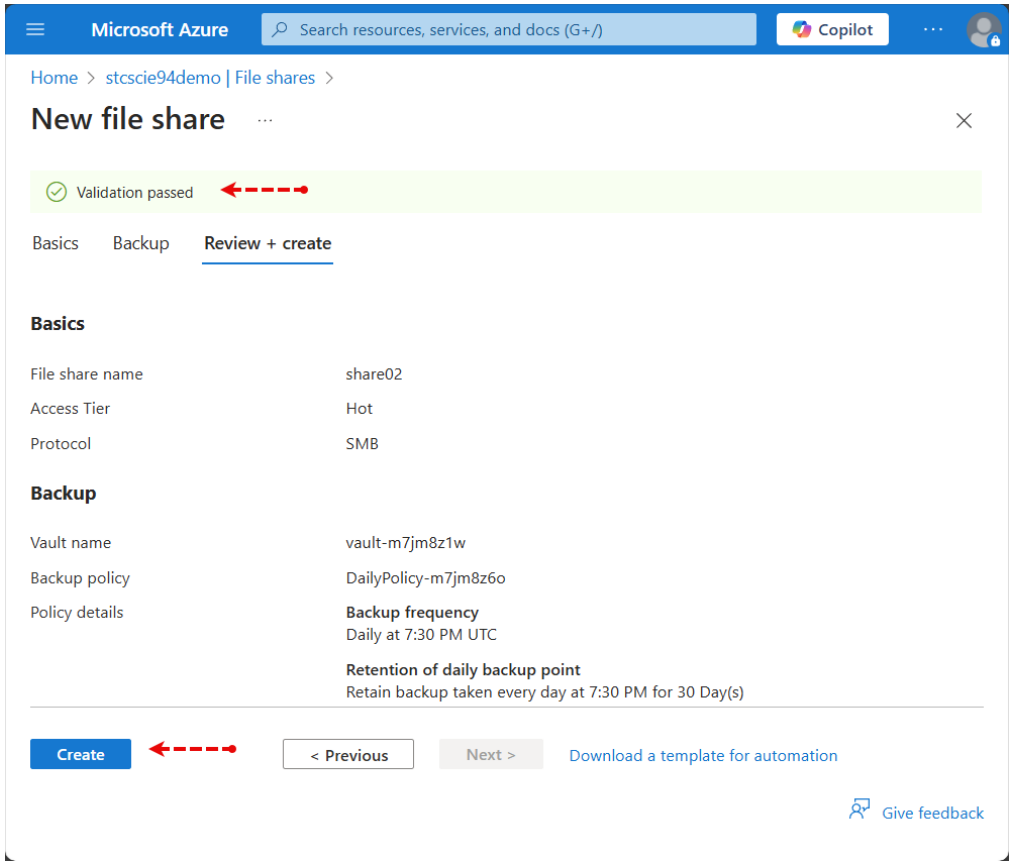
Azure File Storage

- Enable Backup



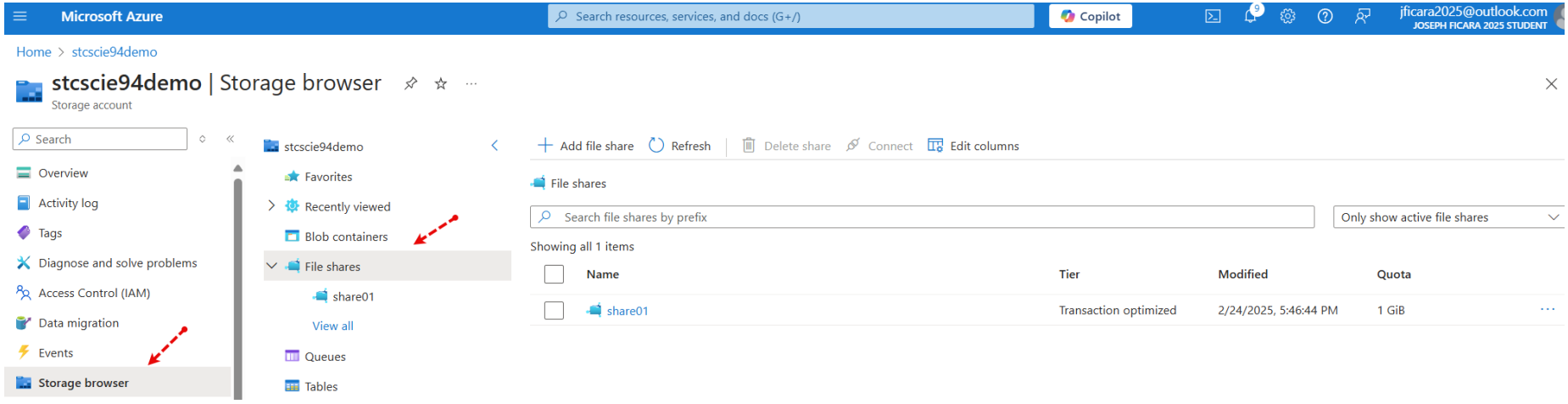
Azure File Storage

- Review and Create



Azure File Storage
Azure Portal - Storage Browser Tables

- The azure portal has a Files browser
  - You need to have
    - **Storage File Data Privileged Contributor**
    - See: Authorize access to Azure file share data



Demo

Azure Storage Files

- Azure Portal

Azure File Storage

- PowerShell is also supported
  - Several cmdlets: http://bit.ly/2Vd5bkv
    - Creating a share
      - New-AzureStorageShare
    - Accessing a share
      - Get-AzureStorageShare
    - Getting Directories and File Listings
      - Get-AzureStorageDirectory
    - Directory creation
      - New-AzureStorageDirectory
    - Upload a file
      - Set-AzureStorageFileContent

Azure File Storage

- Basic steps to upload a file
  - Create a storage context
    - New-AzureStorageContext
      - Provide Storage Account and Key
  - Get a share
    - Get-AzureStorageShare
  - Upload a file
    - Set-AzureStorageFileContent
  - List files in the share/path
    - Get-AzureStorageFile

Azure File Storage

- Note:
  - Get-AzureStorageFile

```
Get-AzureStorageFile –Context $ctx -ShareName $s.Name -Path ClassDemo
```

Azure File Storage

- Install Azure PowerShell Commands
- Launch PowerShell as an Administrator
- Execute the following:
  - `Install-Module -Name Az -AllowClobber`
  - `Install-Module -Name AzureRM –AllowClobber`
- See: (Retiring on 2/29/2025)
  - http://bit.ly/3qrfiPf

Azure File Storage

- You can also map/mount a share
- Validate your configuration first
  - Validation scripts available GitHub
  - azure-files-samples
    - Linux
      - sudo bash AzFileDiagnostics.sh
    - Windows
      - Run from an elevated PowerShell prompt
        - .\AzFileDiagnostics.ps1

Demo

Uploading a file using PowerShell

`AzureFilesDemo.ps1`

Azure File Storage

Windows: Verify & Map a drive letter to an Azure Share

```
# Verify configuration
.\AzFileDiagnostics.ps1

# Mount Azure Share as a drive letter
net use <Drive Letter>: \\<storage account>\<share name>/User:Azure\<storage account name> <storage account key>
```

Example:

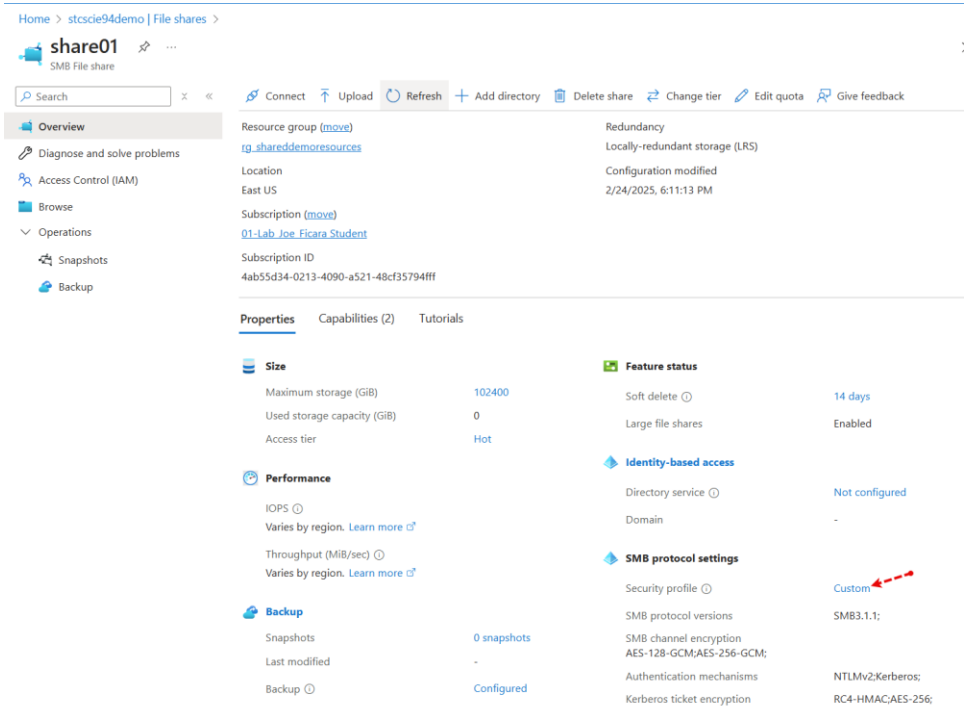
```
net use Q: \\stcscie94demo.file.core.windows.net\share01 /User:Azure\stcscie94demo adiazielsialaeixclsitieidlaixt==
```

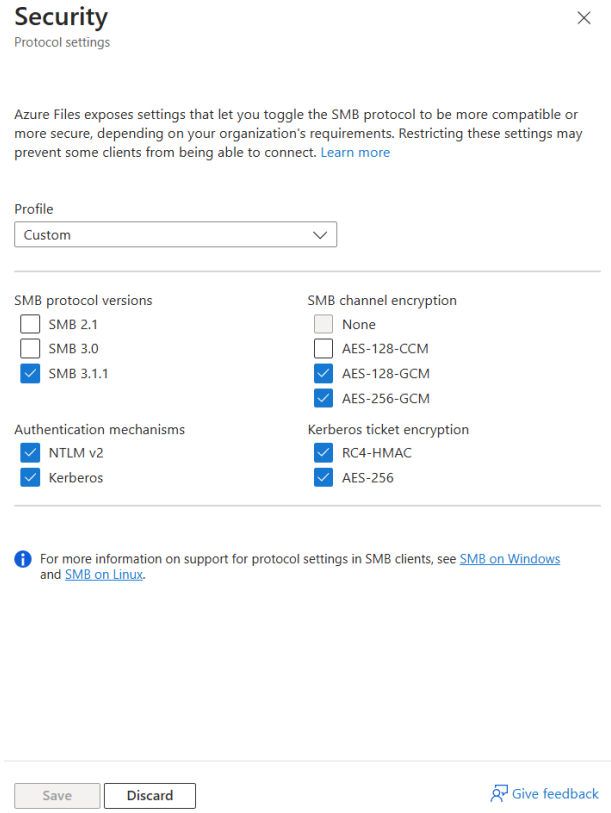
Azure File Storage

- Security settings
  - Use most restrictive possible
  - SMB Storage on Windows
    - Mount Azure file share on Windows
  - SMB Storage on Linux
    - Mount SMB Azure file share on Linux

Azure File Storage

- These work for Ubuntu and Windows 11





Demo

Check pre-req and mount share on Windows

Live Demo

Azure File Storage

Linux: Verify & Mount an Azure Share

```
# Verify configuration
sudo bash AzFileDiagnostics.sh

# Create the mount point directory
sudo mkdir <mount point name>

# Mount an Azure Share
mount -t cifs //<storage account>/<share name>  <mount point name> -o
vers=3.1.1,username=<storage account name>,password='<storage access
key>',dir_mode=0777,file_mode=0777,nosharesock,actimeo=30,sec=ntlmssp
```

Example:

```
sudo mkdir /home/jficara/azureshare01
sudo mount -t cifs //stcscie94storage.file.core.windows.net/share01
azureshare01 -o vers=3.1.1,username=stcscie94storage,password='
adiazielsialaeixclsitieidlaixt==',dir_mode=0777,file_mode=0777,nosharesock,acti
meo=30,sec=ntlmssp
```

Demo

Check pre-req and mount share on Linux

`Live Demo`

Azure File Storage

Links & Resources

- Introduction to Azure Storage
  - Introduction to Azure Storage - Cloud storage on Azure
- Azure Import/Export service
  - Using Azure Import/Export to transfer data to and from Azure Storage
- Azure Storage Scalability & Performance
  - Scalability and performance targets for standard storage accounts - Azure Storage
- Azure subscription & service limits, quotas, & constraints
  - Azure subscription and service limits, quotas, and constraints - Azure Resource Manager

Links & Resources

- How to use:
  - Azure Blob Storage
    - Azure Blob Storage | Microsoft Azure
  - Azure Table Storage
    - Table storage | Microsoft Azure
  - Azure Queues
    - Queue Storage | Microsoft Azure
  - Azure File Storage
    - Azure Files | Microsoft Azure

Links & Resources

- How to use:
  - Storage Service REST APIs
    - Azure Storage REST API Reference | Microsoft Docs
  - Azure Storage PowerShell Cmdlets
    - Install Azure PowerShell with PowerShellGet | Microsoft Docs

Links & Resources

- New SDK:
  - Azure.Data.Tables

Links & Resources

- SDKs
  - .NET
  - Java
  - Go
  - Python
  - Node.JS
  - REST APIs
    - Azure documentation | Microsoft Docs
    - Azure SDK Latest Releases | Azure SDKs