Azure Storage Accounts – Table, Blob, Queue, and File

Reza Salehi
CLOUD CONSULTANT

@zaalion   linkedin.com/in/rezasalehi2008

Overview

Understanding Azure Storage Account

- Table, blob, queue, and file

Understanding Azure Table Storage

- How is the data structured?

Securing Azure Table Storage

- Account Keys, Shared Access Token (SAS), and RBAC

Azure Storage Explorer and storage SDK

Blobs, files, and queues

Summary

Azure Storage Account

Contains all Azure Storage data objects (table, blob, file, queue)

Is accessible from anywhere in the world over HTTP or HTTPS

Is durable and highly available, secure, and massively scalable

Azure Storage Account Types

General-purpose v1

Legacy account type, use general-purpose v2 accounts instead

General-purpose v2

Basic storage account type for blobs, files, queues, and tables

Block blob storage

Blob-only storage accounts with premium performance

FileStorage storage

Files-only storage accounts with premium performance

Blob storage

Blob-only storage, use general-purpose v2 accounts instead

General-purpose V2 Account

Azure Table Storage

Azure Blob Storage

Azure Storage Queues

Azure File Storage

Azure Table Storage

Azure Table Storage

Can store large amounts of data

Is a NoSQL datastore

Is ideal for storing structured, non-relational data

Table Storage Use Cases

Storing TBs of structured data capable of serving web scale applications

For datasets that don't need complex joins, foreign keys, or stored procedures

Allows to quickly query data using a clustered index

Accessing data using the OData protocol and LINQ queries (via the SDK)

"Use Table storage to store and query huge sets of structured, non-relational data, and the tables will scale as demand increases."

Microsoft

Table Storage Concepts

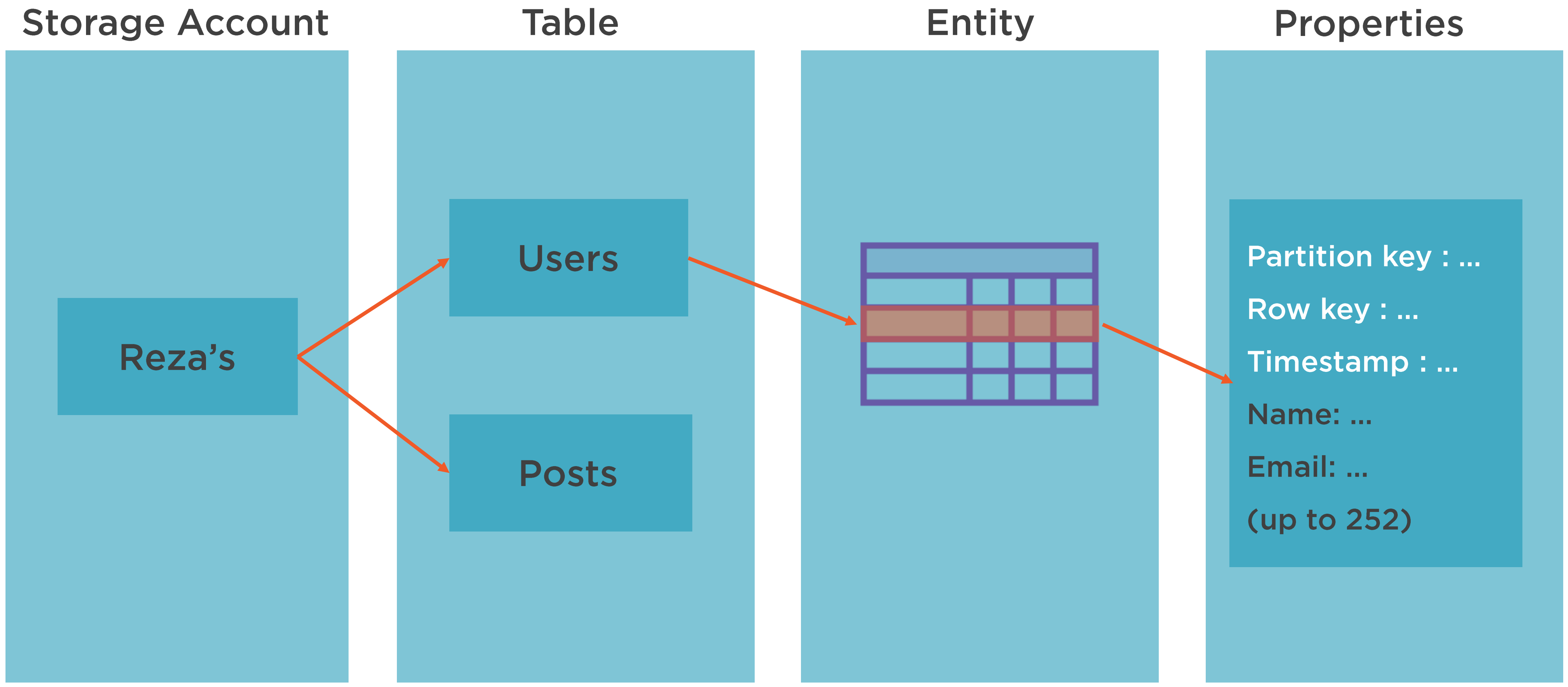


Table Storage Concepts

URL format

http://reza.table.core.windows.net

Account

Access is possible through the storage account

Table

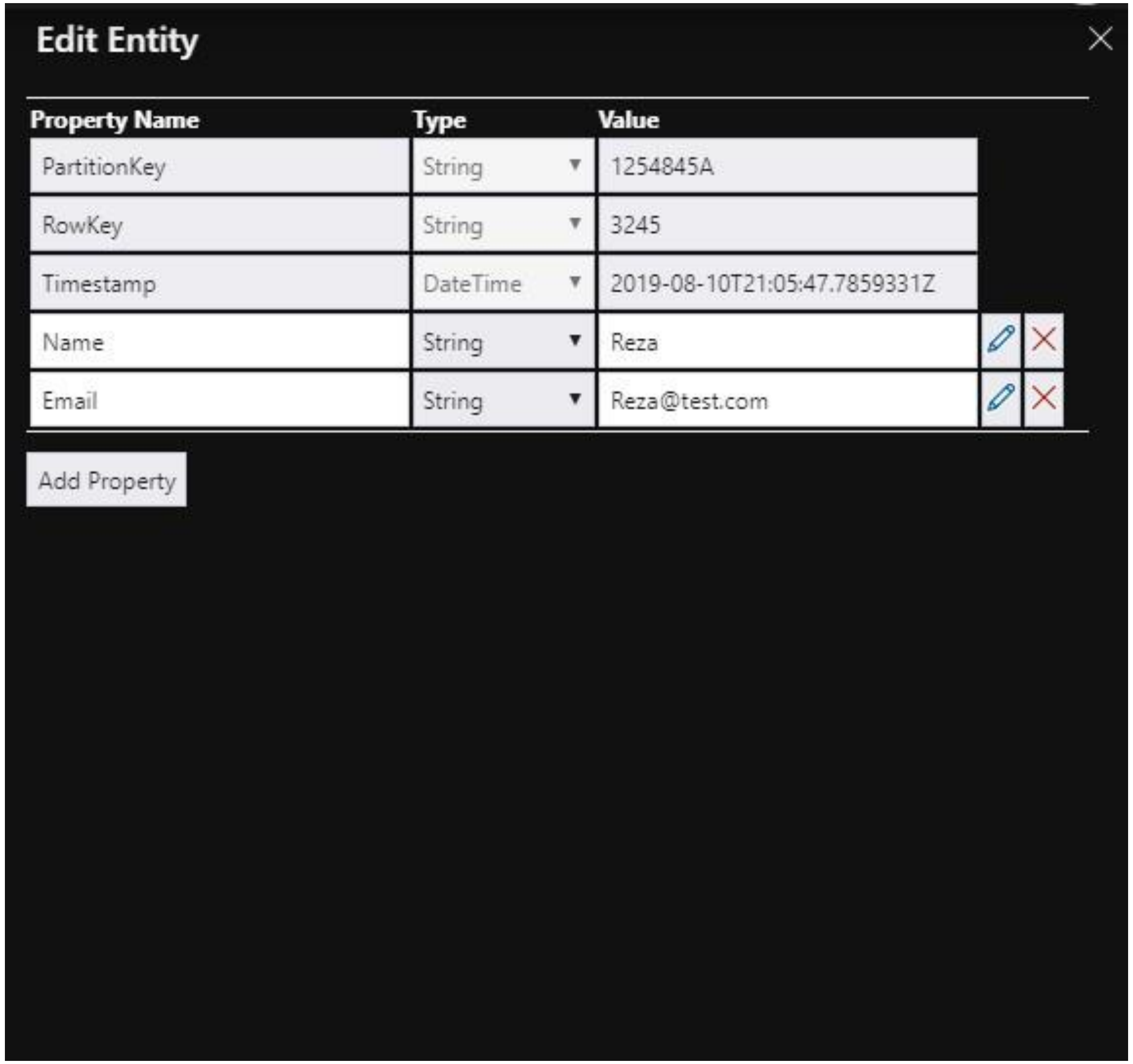
Collection of entities, no schema is enforced

Entity

A set of properties, like a database row

Properties

Name-value pairs



Edit Entity

Property Name	Type	Value
PartitionKey	String	1254845A
RowKey	String	3245
Timestamp	DateTime	2019-08-10T21:05:47.7859331Z
Name	String	Reza
Email	String	Reza@test.com

Properties

A property is a name-value pair

Each entity can include up to 252 properties to store data

Also, three system properties specify a partition key, a row key, and a timestamp

Query entities with same partition key more quickly and insert/update them in atomic operations

Table Storage System Properties

PartitionKey

The developer is responsible for inserting and updating its value

RowKey

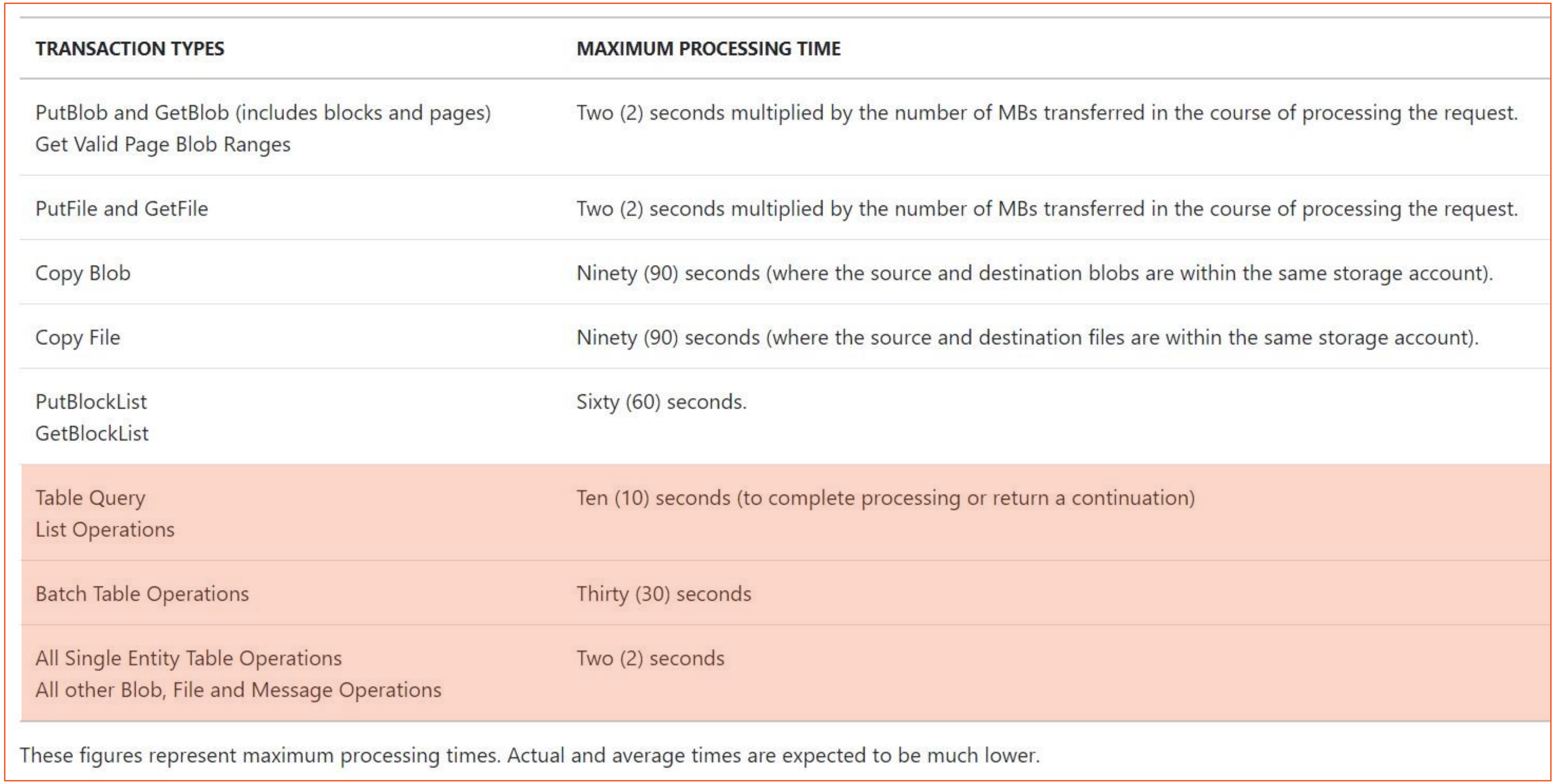
The developer is responsible for inserting and updating its value

Timestamp

The server manages its value & it cannot be manually modified

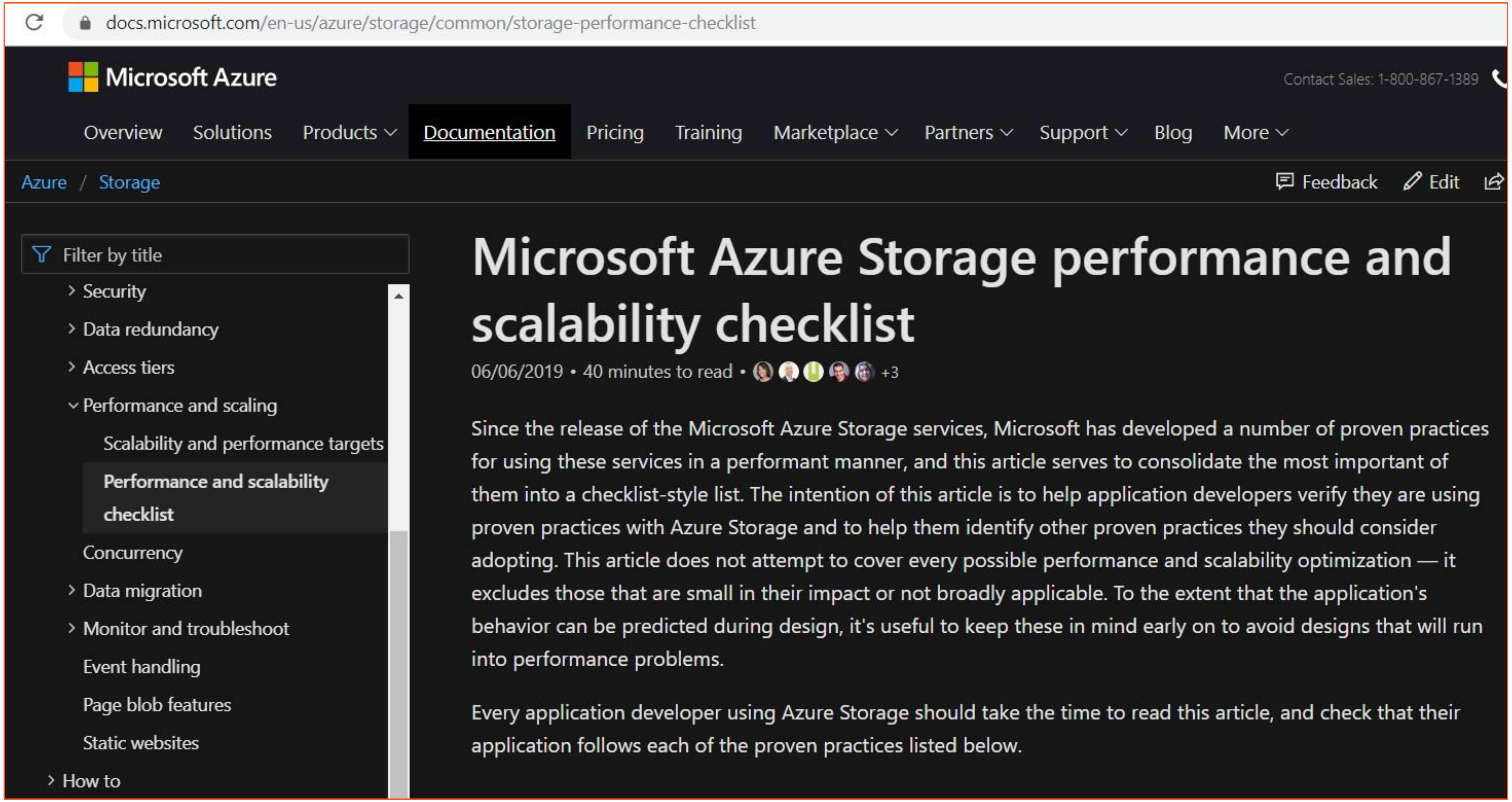
An entity in Azure Storage can be up to 1MB in size.

SLA for Storage Accounts



TRANSACTION TYPES	MAXIMUM PROCESSING TIME
PutBlob and GetBlob (includes blocks and pages) Get Valid Page Blob Ranges	Two (2) seconds multiplied by the number of MBs transferred in the course of processing the request.
PutFile and GetFile	Two (2) seconds multiplied by the number of MBs transferred in the course of processing the request.
Copy Blob	Ninety (90) seconds (where the source and destination blobs are within the same storage account).
Copy File	Ninety (90) seconds (where the source and destination files are within the same storage account).
PutBlockList GetBlockList	Sixty (60) seconds.
Table Query List Operations	Ten (10) seconds (to complete processing or return a continuation)
Batch Table Operations	Thirty (30) seconds
All Single Entity Table Operations All other Blob, File and Message Operations	Two (2) seconds

These figures represent maximum processing times. Actual and average times are expected to be much lower.



Microsoft Azure Storage performance and scalability checklist

06/06/2019 • 40 minutes to read • +3

Since the release of the Microsoft Azure Storage services, Microsoft has developed a number of proven practices for using these services in a performant manner, and this article serves to consolidate the most important of them into a checklist-style list. The intention of this article is to help application developers verify they are using proven practices with Azure Storage and to help them identify other proven practices they should consider adopting. This article does not attempt to cover every possible performance and scalability optimization — it excludes those that are small in their impact or not broadly applicable. To the extent that the application's behavior can be predicted during design, it's useful to keep these in mind early on to avoid designs that will run into performance problems.

Every application developer using Azure Storage should take the time to read this article, and check that their application follows each of the proven practices listed below.

Azure Cosmos DB Table API
is the premium offering for
Azure Table Storage.

Blob, Queue, and File Storage

"Azure Blob storage is optimized to store massive amounts of unstructured data in the cloud."

Microsoft

Azure Blob Storage Use Cases

- Serving documents directly to a browser
- Streaming video and audio
- Storing log files
- Storing data for backup and restore
- Storing data for analysis
- Storing files for distributed access

"Azure Queue storage is a service for storing large numbers of messages. Access messages via authenticated calls using HTTP or HTTPS."

Microsoft

Azure Queue Storage Use Cases

Creating a backlog of work to process asynchronously

Passing messages from a web role to a worker role

"Azure Files offers fully managed file shares in the cloud that are accessible via the industry standard Server Message Block (SMB) protocol."

Microsoft

Azure Files Use Cases

Replace on-premises file servers

"Lift and shift" scenarios

Securing Storage Accounts

Azure Storage Security

Management Security

Encryption for data in transit

Audit/monitor access

Data access security

Encryption for data at rest

CORS for browser clients

Storage Account Management Security

Operations that affect the storage account itself (create or delete a storage account)

RBAC roles for storage (owner, contributor, reader, etc.)

Assigning the appropriate RBAC role to users, groups or applications

Storage Account Data Access Security

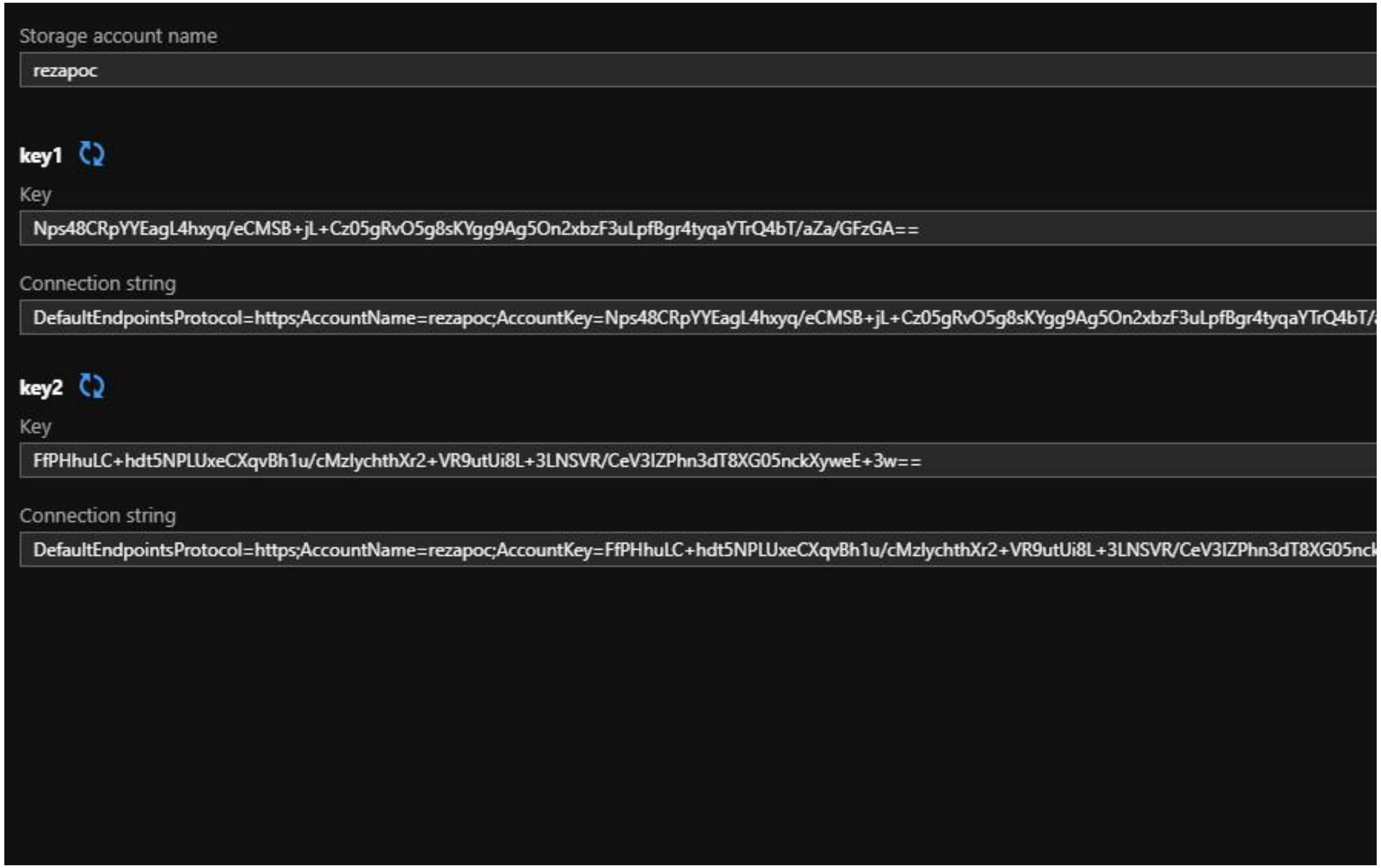
Storage Account Keys

It grants complete access

Shared Access Signatures (SAS)

Give the permissions required for a limited amount of time

Storage Account Keys



Storage account name

rezapoc

key1

Key

```
Nps48CRpYYEagL4hxyq/eCMSB+jL+Cz05gRvO5g8sKYgg9Ag5On2xbzF3uLpfBgr4tyqaYTrQ4bT/aZa/GFzGA==
```

Connection string

```
DefaultEndpointsProtocol=https;AccountName=rezapoc;AccountKey=Nps48CRpYYEagL4hxyq/eCMSB+jL+Cz05gRvO5g8sKYgg9Ag5On2xbzF3uLpfBgr4tyqaYTrQ4bT/
```

key2

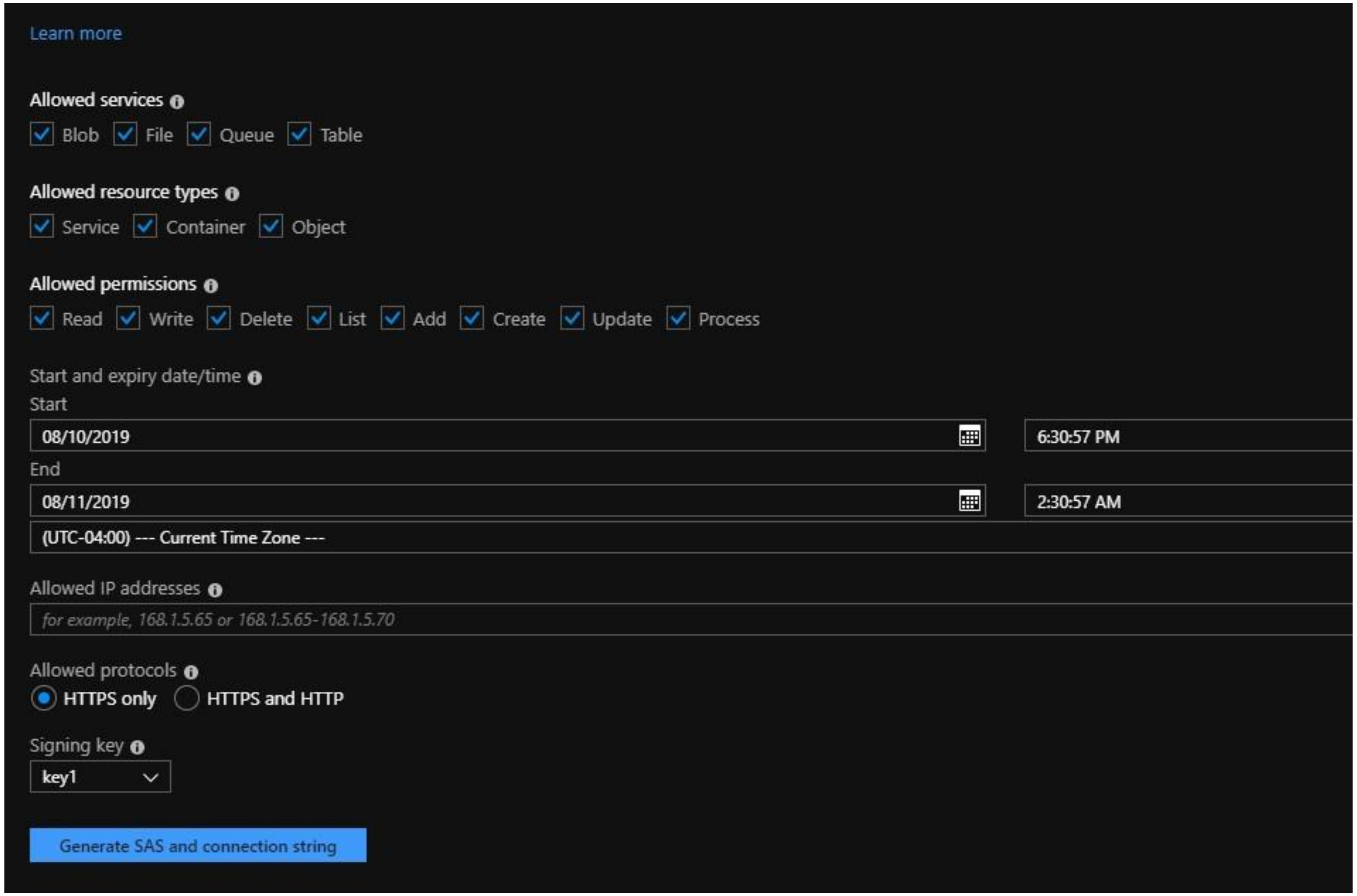
Key

```
FfPHhuLC+hdt5NPLUxeCXqvBh1u/cMzlychthXr2+VR9utUi8L+3LNSVR/CeV3IZPhn3dT8XG05nckXyweE+3w==
```

Connection string

```
DefaultEndpointsProtocol=https;AccountName=rezapoc;AccountKey=FfPHhuLC+hdt5NPLUxeCXqvBh1u/cMzlychthXr2+VR9utUi8L+3LNSVR/CeV3IZPhn3dT8XG05nc
```

Shared Access Signatures (SAS)



Learn more

Allowed services
[✓ Blob] [✓ File] [✓ Queue] [✓ Table]

Allowed resource types
[✓ Service] [✓ Container] [✓ Object]

Allowed permissions
[✓ Read] [✓ Write] [✓ Delete] [✓ List] [✓ Add] [✓ Create] [✓ Update] [✓ Process]

Start and expiry date/time

Start
08/10/2019 6:30:57 PM

End
08/11/2019 2:30:57 AM

(UTC-04:00) --- Current Time Zone ---

Allowed IP addresses
for example, 168.1.5.65 or 168.1.5.65-168.1.5.70

Allowed protocols
[✓ HTTPS only] HTTPS and HTTP

Signing key
key1

Generate SAS and connection string

Encryption for Data in Transit

Always use HTTPS when calling the REST APIs or accessing objects in storage

SAS tokens include an option to enforce HTTPS protocol for the token sender

Encryption for Data at Rest

Storage Service Encryption (SSE)

Is enabled for all storage accounts and cannot be disabled

Client-side Encryption

Programmatically encrypt the data in a client application then send it across the wire

Audit/Monitor Access

You can enable Azure Storage Analytics to log the authentication method used by clients.

CORS for Browser-based Clients

When a web browser running in one domain makes an HTTP request for a different domain, this is called a cross-origin HTTP request

When calling Azure Table Storage API that return JSON data to be processed by the JavaScript client

You can allow origins, methods, and headers

Demo

Provisioning Table Storage in the Azure Portal

- Configuring security

Demo

Working with Azure Storage Explorer

Demo

Working with the Azure Table Storage .NET SDK

Demo

Provisioning other storage types

- Blobs
- Queues
- Files

Summary

Understanding Azure Storage Account

- Table, blob, queue and file

Understanding Azure Table Storage

- How is the data structured?

Securing Azure Table Storage

- Account Keys, SAS tokens & RBAC and more

Azure Storage Explorer & storage SDK

Blobs, files and queues