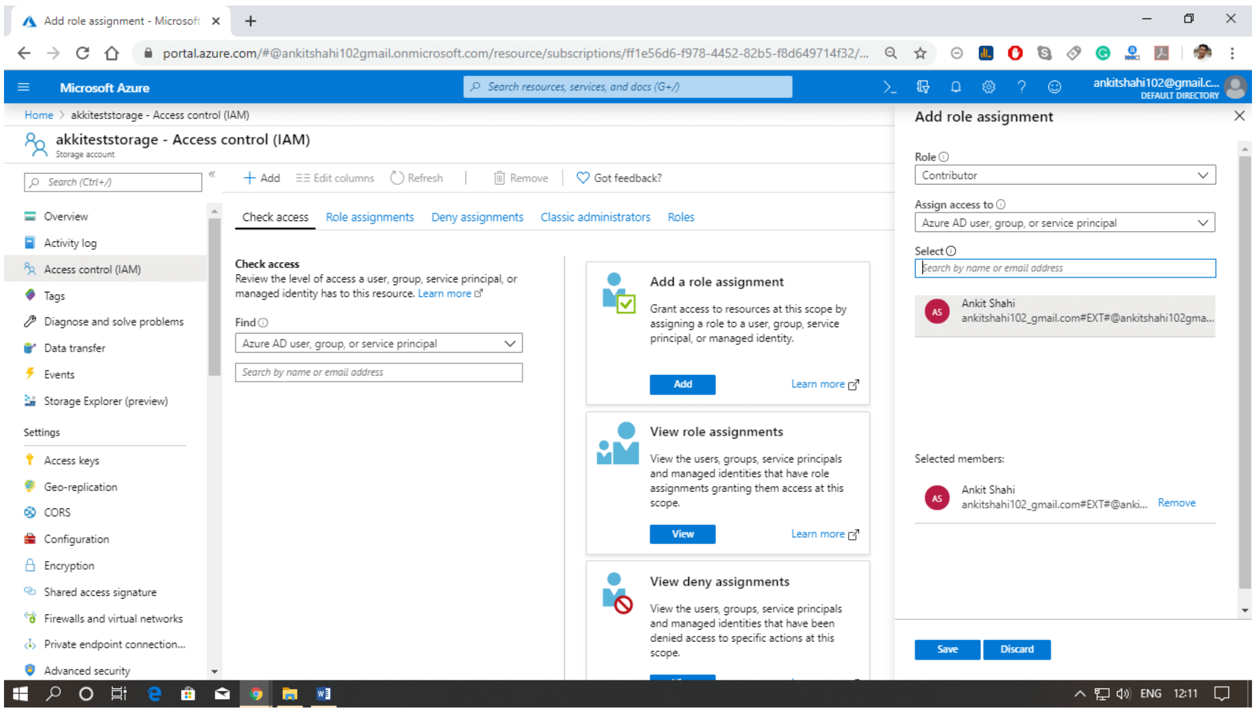
Azure Storage Security

Azure storage security is divided into five major areas.

Management plane security

The management plane refers to the operation that affects the storage account itself. The way we control access to the services that affect the storage account is by using Azure active directory.



Check access
Review the level of access a user, group, service principal, or managed identity has to this resource. Learn more

Find: Azure AD user, group, or service principal
Search by name or email address

Add a role assignment
Grant access to resources at this scope by assigning a role to a user, group, service principal, or managed identity.

View role assignments
View the users, groups, service principals and managed identities that have role assignments granting them access at this scope.

View deny assignments
View the users, groups, service principals and managed identities that have been denied access to specific actions at this scope.

Add role assignment

Role: Contributor

Assign access to: Azure AD user, group, or service principal

Select: Search by name or email address

Ankit Shahi
ankitshahi102_gmail.com#EXT#@ankitshahi102gma...

Selected members:

Ankit Shahi
ankitshahi102_gmail.com#EXT#@anki... Remove

Role-based access control

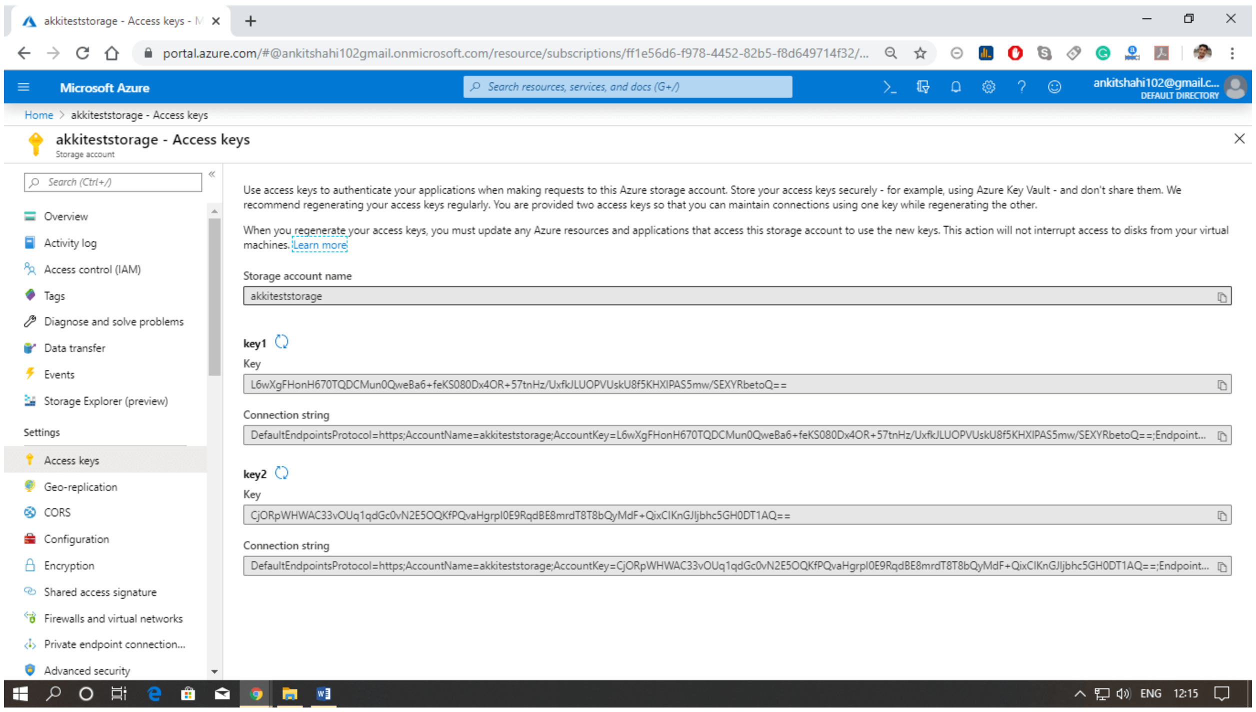
- As we are aware that every Azure subscription has an associated Azure active directory. The Azure active directory contains users, groups, and applications. To them, we can provide access to manage resources within the Azure subscription. That resource can be a storage account, and the way we control the level of access to storage accounts is by assigning an appropriate role to the user. So we can have an owner role or contributor role or reader role that we can define.

Key Points to remember:

- When we are assigning a role, we can control access to the operations used to manage the storage account but not data objects in the account.
- However, we can give access to data objects by providing permission to read storage account keys because storage account keys enable the users to have access to data objects.
- Each role has a list of actions.
- There are some standard roles available, e.g., Owner, Reader, Contributor, etc.
- We can define a new custom role by selecting a set of actions from the list of available actions.

Data Plane security

It refers to the methods used to secure data objects (blobs, queues, tables, and files) within the storage account.


akkiteststorage - Access keys - M
portal.azure.com/#@ankitshahi102gmail.onmicrosoft.com/resource/subscriptions/ff1e56d6-f978-4452-82b5-f8d649714f32/...
Microsoft Azure
Search resources, services, and docs (G+/)
ankitshahi102@gmail.c...
DEFAULT DIRECTORY
Home > akkiteststorage - Access keys
akkiteststorage - Access keys
Storage account
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Data transfer
Events
Storage Explorer (preview)
Settings
Access keys
Geo-replication
CORS
Configuration
Encryption
Shared access signature
Firewalls and virtual networks
Private endpoint connection...
Advanced security
ENG
12:15

Use access keys to authenticate your applications when making requests to this Azure storage account. Store your access keys securely - for example, using Azure Key Vault - and don't share them. We recommend regenerating your access keys regularly. You are provided two access keys so that you can maintain connections using one key while regenerating the other.

When you regenerate your access keys, you must update any Azure resources and applications that access this storage account to use the new keys. This action will not interrupt access to disks from your virtual machines. Learn more

Storage account name: akkiteststorage

key1

Key: L6wXgFHonH670TQDCMun0QweBa6+feKS080Dx4OR+57tnHz/UxfkJLUOPVUskU8f5KHXIPAS5mw/SEXYRbetoQ==

Connection string: DefaultEndpointsProtocol=https;AccountName=akkiteststorage;AccountKey=L6wXgFHonH670TQDCMun0QweBa6+feKS080Dx4OR+57tnHz/UxfkJLUOPVUskU8f5KHXIPAS5mw/SEXYRbetoQ==;Endpoint...

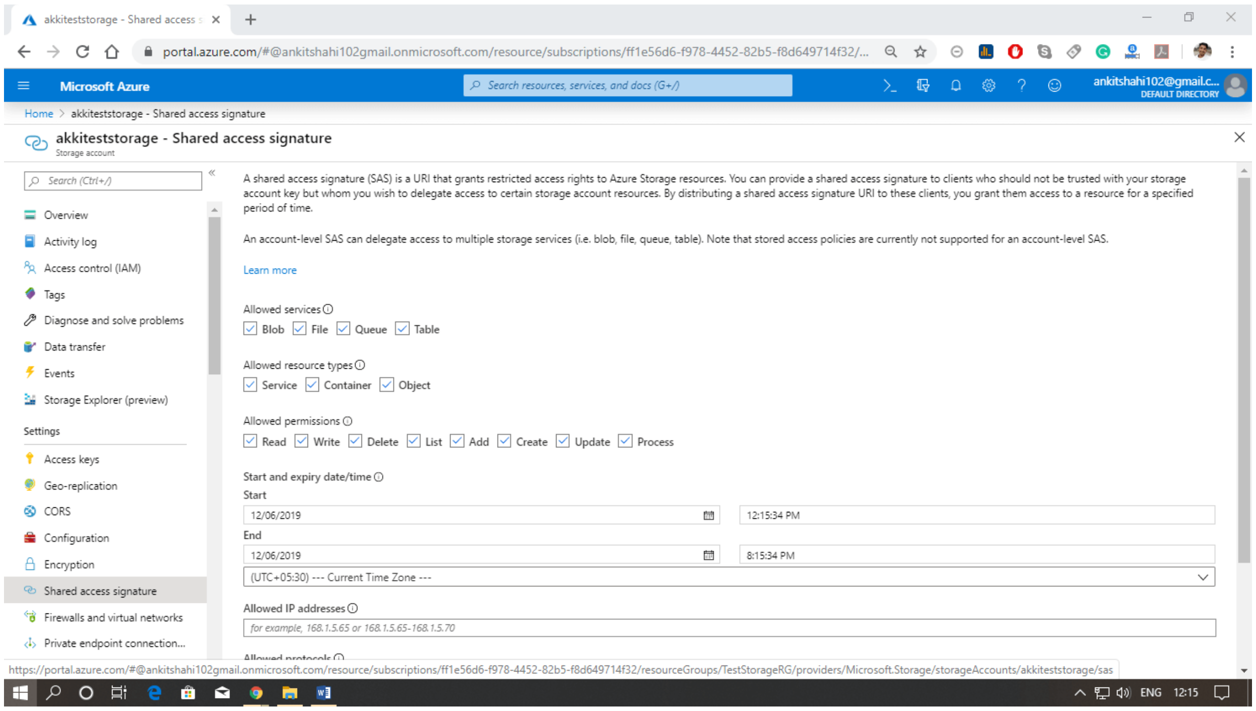
key2

Key: CjORpWHWAC33vOUq1qdGc0vN2E5OQKfPQvaHgrpl0E9RqdBE8mrdT8T8bQyMdF+QixCIKnGJljbhc5GH0DT1AQ==

Connection string: DefaultEndpointsProtocol=https;AccountName=akkiteststorage;AccountKey=CjORpWHWAC33vOUq1qdGc0vN2E5OQKfPQvaHgrpl0E9RqdBE8mrdT8T8bQyMdF+QixCIKnGJljbhc5GH0DT1AQ==;Endpoint...

There are three ways that you can control access to the data within the storage account

- ***Azure active directory*** authorizes access to containers and queues. Azure Active Directory provides advantages over other approaches to authorization, including removing the need to store secrets in your code.
- ***Storage account keys*** provide blanket access to all data objects within the storage account.
- ***Shared Access Signatures***, in case, if we want to provide access to certain services, for example - only to blobs, only to queues, or a combination of them. And also, if we want to control the level of access, for example - read-only, update, delete in that way, and also we wish to provide time-limited access. So we want to give access to only one year, and after that one year, we generate another SAS and present it to them for security reasons. In that case, we use shared access signatures.


akkiteststorage - Shared access
portal.azure.com/#@ankitshahi102gmail.onmicrosoft.com/resource/subscriptions/ff1e56d6-f978-4452-82b5-f8d649714f32/...
Microsoft Azure
Search resources, services, and docs (G+/)
ankitshahi102@gmail.c...
DEFAULT DIRECTORY
Home > akkiteststorage - Shared access signature
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Data transfer
Events
Storage Explorer (preview)
Settings
Access keys
Geo-replication
CORS
Configuration
Encryption
Shared access signature
Firewalls and virtual networks
Private endpoint connection...
Learn more
ENG
12:15

akkiteststorage - Shared access signature
Storage account

A shared access signature (SAS) is a URI that grants restricted access rights to Azure Storage resources. You can provide a shared access signature to clients who should not be trusted with your storage account key but whom you wish to delegate access to certain storage account resources. By distributing a shared access signature URI to these clients, you grant them access to a resource for a specified period of time.

An account-level SAS can delegate access to multiple storage services (i.e. blob, file, queue, table). Note that stored access policies are currently not supported for an account-level SAS.

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
12/06/2019 12:15:34 PM
End
12/06/2019 8:15:34 PM
(UTC+05:30) --- Current Time Zone ---

Allowed IP addresses
for example, 168.1.5.65 or 168.1.5.65-168.1.5.70

https://portal.azure.com/#@ankitshahi102gmail.onmicrosoft.com/resource/subscriptions/ff1e56d6-f978-4452-82b5-f8d649714f32/resourceGroups/TestStorageRG/providers/Microsoft.Storage/storageAccounts/akkiteststorage/sas

We can allow public access to our blobs by setting the access level for the container that holds the blob accordingly.

Encryption in transit

Transport level Encryption using HTTPS

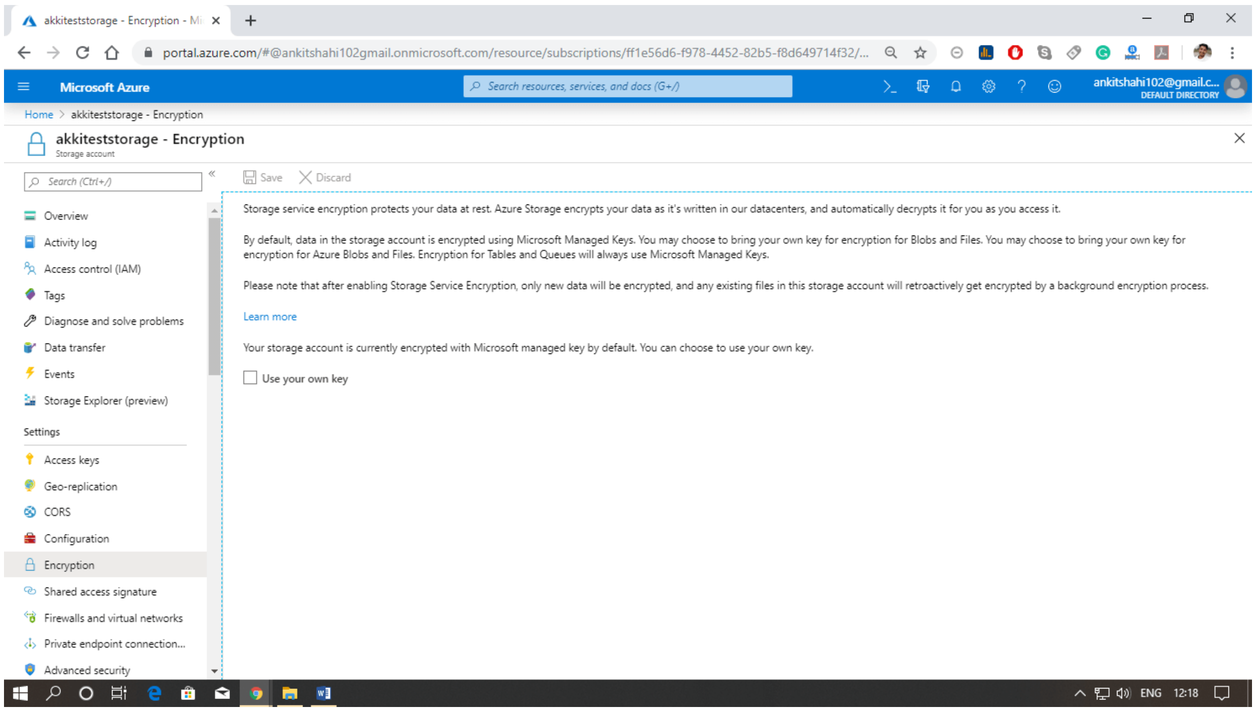
- Always use HTTPS when using REST APIs or accessing the object in storage.
- If we are using SAS, we can specify that only HTTPS should be applied.

Using encryption in transit for Azure file shares

- 1 does not support encryption, so connections are only allowed within the same region.
- 0 supports encryption, and cross-region access is allowed.

Client-side encryption

- Encrypt the data before being transferred to Azure storage
- When retrieving the data form Azure, data is decrypted after it is received on the client-side.


akkiteststorage - Encryption - Mi
portal.azure.com/#@ankitshahi102gmail.onmicrosoft.com/resource/subscriptions/ff1e56d6-f978-4452-82b5-f8d649714f32/...
Microsoft Azure
Search resources, services, and docs (G+/)
ankitshahi102@gmail.c...
DEFAULT DIRECTORY
Home > akkiteststorage - Encryption
akkiteststorage - Encryption
Storage account
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Data transfer
Events
Storage Explorer (preview)
Settings
Access keys
Geo-replication
CORS
Configuration
Encryption
Shared access signature
Firewalls and virtual networks
Private endpoint connection...
Advanced security
Save
Discard
ENG
12:18

Storage service encryption protects your data at rest. Azure Storage encrypts your data as it's written in our datacenters, and automatically decrypts it for you as you access it.

By default, data in the storage account is encrypted using Microsoft Managed Keys. You may choose to bring your own key for encryption for Blobs and Files. You may choose to bring your own key for encryption for Azure Blobs and Files. Encryption for Tables and Queues will always use Microsoft Managed Keys.

Please note that after enabling Storage Service Encryption, only new data will be encrypted, and any existing files in this storage account will retroactively get encrypted by a background encryption process.

Learn more

Your storage account is currently encrypted with Microsoft managed key by default. You can choose to use your own key.

☐ Use your own key

Encryption at rest

Client-side encryption

- Encrypt the data before being transferred to Azure storage.
- When retrieving the data form Azure, data is decrypted after it is received on the client-side.

Storage Service Encryption (SSE)

This is what we generally use to encrypt the data at REST is Azure storage

- It is enabled for all storage accounts and cannot be disabled.
- It automatically encrypts data in all performance tiers (Standard and premium), all deployment models (Azure Resource Manager and Classic), and all of the Azure Storage services (Blob, Queue, Table, and File). So it is blanket encryption across all Azure storage.
- We can use either Microsoft-managed keys or your custom keys to encrypt the data.

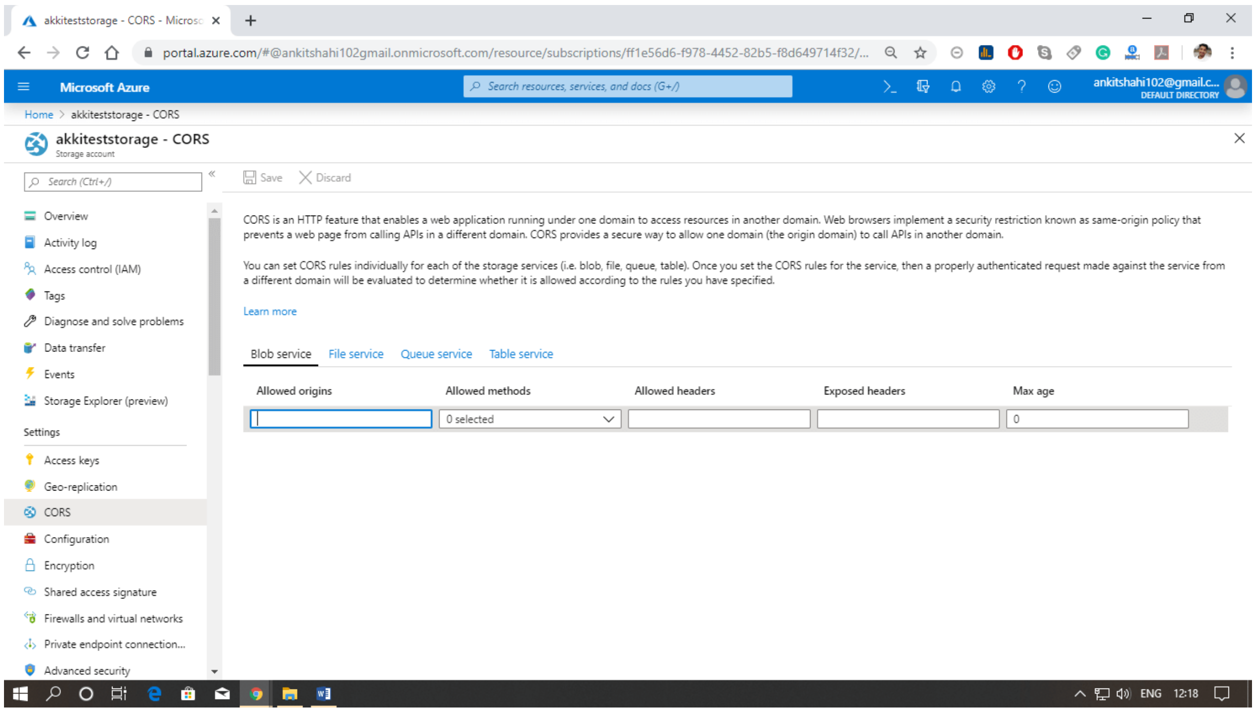
Azure Disk Encryption

This is a recommended approach from Microsoft to encrypt the disks particularly with Azure disk

- Encrypt the OS & data disks used by IaaS Virtual Machine
- You can enable encryption on existing IaaS VMs
- You can use customer-provided encryption keys

CORS (Cross-Origin Resource Sharing)

- When a web browser makes an HTTP request for a resource from a different domain, this is called a cross-origin HTTP request.
- Azure Storage allows us to enable CORS. For each storage account, we can specify domains that can access the resources in that storage account. For example, enable CORS on the mystorage.blob.core.windows.net storage account and configure it to allow access to mywebsite.com.
- CORS allows access but does not provide authentication, which means we still need to use SAS keys to access non-public storage resources.
- CORS is disabled on all services by default. We can enable it using the Azure portal or Power Shell, and we can specify the domains from where the request will come to access the data in your storage account.


akkiteststorage - CORS - Microso
portal.azure.com/#@ankitshahi102gmail.onmicrosoft.com/resource/subscriptions/ff1e56d6-f978-4452-82b5-f8d649714f32/...
Microsoft Azure
Search resources, services, and docs (G+/)
ankitshahi102@gmail.c...
DEFAULT DIRECTORY
Home > akkiteststorage - CORS
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Data transfer
Events
Storage Explorer (preview)
Settings
Access keys
Geo-replication
CORS
Configuration
Encryption
Shared access signature
Firewalls and virtual networks
Private endpoint connection...
Advanced security
Save
Discard
ENG
12:18

akkiteststorage - CORS

Storage account

CORS is an HTTP feature that enables a web application running under one domain to access resources in another domain. Web browsers implement a security restriction known as same-origin policy that prevents a web page from calling APIs in a different domain. CORS provides a secure way to allow one domain (the origin domain) to call APIs in another domain.

You can set CORS rules individually for each of the storage services (i.e. blob, file, queue, table). Once you set the CORS rules for the service, then a properly authenticated request made against the service from a different domain will be evaluated to determine whether it is allowed according to the rules you have specified.

Learn more

Blob service | File service | Queue service | Table service

Allowed origins	Allowed methods	Allowed headers	Exposed headers	Max age
	0 selected			0