Service Endpoint Policies

Last updated by | Vitor Tomaz | Jun 8, 2022 at 5:38 AM PDT

Contents

- Training about the feature
- Self-help content presented in Azure Portal
  - Learn how service endpoint policies help prevent data exfil...
  - How service endpoint policies help prevent data exfiltration
  - Service endpoints default to denying connections to Azure...
  - When to use service endpoints and policies in Azure SQL ...
  - Configure service endpoint policies for Azure SQL Manage...
  - Resources
- Introduction
- Basic concepts for Service Endpoint Policies on SQL MI
- Azure Support Center

Training about the feature

Harden your Azure SQL Managed Instance against data exfiltration

Self-help content presented in Azure Portal

(This content was shown to the customer during case submission. It's also visible on 'Diagnose and solve problems' blade.)

Learn how service endpoint policies help prevent data exfiltration

Virtual Network (VNet) Azure Storage service endpoint policies let you filter egress virtual network traffic to Azure Storage, restricting data transfers to specific storage accounts. The ability to configure your endpoint policies and associate them with your SQL Managed Instance is currently in preview.

How service endpoint policies help prevent data exfiltration

Azure provides multiple mechanisms to secure traffic to and from virtual networks. One way you can increase the security is by attaching service endpoints to your subnet. Service endpoints do several useful things: they route your traffic through fast and secure Azure backbone; apply filtering rules before traffic leaves your subnet; and allow the Azure service being contacted to recognize where the traffic is coming from. More than a dozen Azure services support service endpoints. Azure Storage is one of those.

In other words, with an Azure Storage service endpoint in place on our Azure SQL Managed Instance subnet, you can establish three security mechanisms:

1. Traffic between your databases and Azure Storage is now carried by Azure’s backbone. 1.Your storage accounts can reject connections that don’t originate from your subnets.
2. Azure SQL Managed Instance subnet can stop services from reaching out to unsafe storage accounts.

Mechanism 1 is once a service endpoint is in place. Mechanism 2 happens when you configure your storage account for limited public connectivity. For the third mechanism, you need service endpoint policies.

Service endpoints default to denying connections to Azure Storage

Be careful when configuring service endpoint policies because these policies introduce a major change in how a subnet behaves. When you associate your first service endpoint policy with a subnet, all outbound connections to storage accounts will be rejected by default unless the policies allow them. We recommend that you take a cautious approach by first allowing your subnet to access all storage accounts in your subscriptions. Then, narrow that down to resource groups where your storage accounts are. Finally, update your policies one last time to only allow the individual storage accounts. Keep verifying that your workflows work as intended, as you go.

If you’ve been following Azure SQL Managed Instance, you may ask: what about the remote storage that it uses for regular operation and backups? Should I worry about cutting them off with my own policies? The answer is no. Managed Instance will take care to automatically allow all built-in storage accounts with a set of special exceptions that cannot be disabled. This way, you can be confident that your databases will remain operational and backed up regardless of policies on the Managed Instance’s subnet.

When to use service endpoints and policies in Azure SQL Managed Instance

We recommend using service endpoints and policies in Azure SQL Managed Instance as often as possible, especially because there are no extra costs associated with them. Still, there are several common scenarios that involve data flows between Azure SQL Managed Instance and Azure Storage accounts:

- Azure DMS offline migration
- Log Replay Service migration
- When you import data with `BULK INSERT` or `OPENROWSET(BULK ...)` , do a copy-only backup with TO `URL =`, or restore a database from Azure Storage
- If you log extended events to an Event File target, or have your database fire off auditing events to a storage account
- When you synchronize tables using transactional replication, which requires a storage account share

If you use, or plan to use, workflows where Azure Storage is contacted from inside the Azure SQL Managed Instance subnet, consider adding service endpoints to your subnets and securing both ends.

Configure service endpoint policies for Azure SQL Managed Instance

See key benefits, limitations, and how to Configure service endpoint policies for Azure SQL Managed Instance ↗ .

Resources

- Harden your Azure SQL Managed Instance workloads against data exfiltration ↗
- Configure service endpoint policies for Azure SQL Managed Instance ↗

Introduction

Service Endpoint Policy (SEP) is a list of resources on one or more service endpoints (Microsoft.Storage, Eventhub...).
When SEP is associated to the subnet the policy allows access only to these specified resources from the subnet. Therefore we could not initially allow this anti-data-exfiltration feature for customers, because customer SEPs would block our service access to our internal resources on these service endpoints (e.g. remote storage account, backup storage account etc.).
In order to allow the feature (for the time being only for storage endpoint), we had to introduce internal SEPs with our internal storage resources.
Internal SEPs are associated to customer subnet as contextual, meaning they are activated only if/when the customer associates their own customer SEP to the subnet.

For the simplicity in the rest of this TSG, "SEP" refers to "internal SEP" only, not to customer SEP.
NRP naming convention for internal SEP is to start with **_e41f87a2_** prefix.

CMS views of SEP state machines are:

View name	Description
service_endpoint_policies	Internal SEP For MI service, each Managed Instance and each Private Cluster has its own SEP. Name of MI SEP follows the pattern: *e41f87a2_mi*<managed_server_id> (e.g. _e41f87a2_mi_e24bb011-588c-4ba1-b757-f23164a31506) Name of PDC SEP follows the pattern: *e41f87a2_pdc*<private_cluster_id> (e.g. _e41f87a2_pdc_84f11ec1-f20d-4269-99e8-2941f7994d65)
service_endpoint_policy_definitions	Internal SEP definition (list of resources on SE) per service endpoint. For the time being Managed Instance service allows (through NIP) only Microsoft.Storage i.e. there is one definition per each SEP
service_endpoint_policy_management	Contextual SEP - one per customer subnet Associates internal SEPs of our objects in the subnet (managed servers, private clusters) to customer subnet as contextual policies

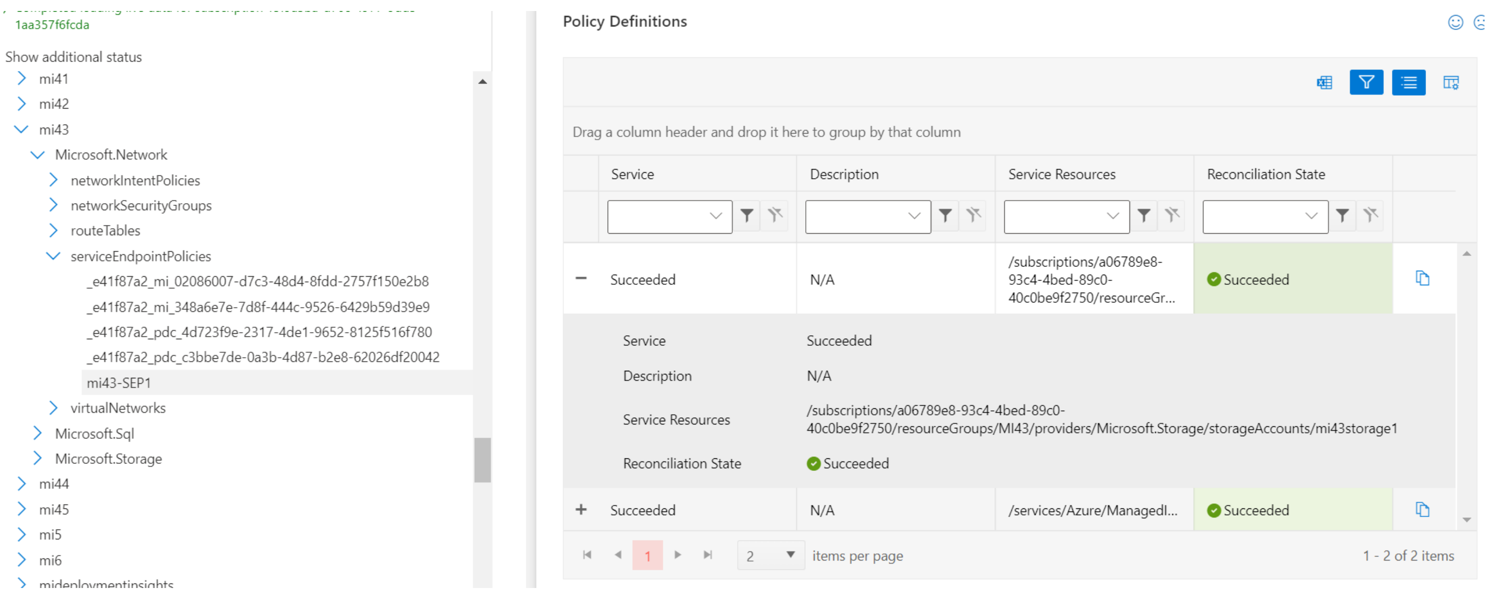
Basic concepts for Service Endpoint Policies on SQL MI

- Internal SEPs are provisioned one per each "owner" (MI, PDC). Owner's internal resources (storage accounts on Microsoft.Storage endpoint) are listed in its accompanying internal SEP.
- A stable MI subnet (no on-going provisioning/upgrading) should have as many Ready internal SEPs as there are MIs and PDCs in the subnet

- Internal SEP name contains *mi* or *pdc* (depending of the owner type: MI or PDC FSM) followed by the ID of the owner FSMs (managed_server_id or private_cluster_id)
- Contextual SEP is a hidden child under the subnet resource that lists all internal SEPs that are attached to the subnet
- Internal SEP that is attached to the subnet is being active only if there are customer (CX) SEPs attached to the subnet. "SEPs being active on the subnet" means that they allow the access from the subnet only to storage accounts listed in SEPs (internal, CX) that are attached to the subnet".
- Internal SEPs are provisioned/decommissioned on create/drop path of their owners (MI, PDC) CX is enabled or disabled to attach their own SEPs on the subnet (with MI) through NIP

Azure Support Center

All the service endpoint policies are visible in ASC Resource Explorer under **serviceEndpointPolicies** that is available on **Microsoft.Network** resource provider.



1aa357f6fcda

Show additional status

- mi41
- mi42
- mi43
  - Microsoft.Network
    - networkIntentPolicies
    - networkSecurityGroups
    - routeTables
    - serviceEndpointPolicies
      - _e41f87a2_mi_02086007-d7c3-48d4-8fdd-2757f150e2b8
      - _e41f87a2_mi_348a6e7e-7d8f-444c-9526-6429b59d39e9
      - _e41f87a2_pdc_4d723f9e-2317-4de1-9652-8125f516f780
      - _e41f87a2_pdc_c3bbe7de-0a3b-4d87-b2e8-62026df20042
      - mi43-SEP1
    - virtualNetworks
  - Microsoft.Sql
  - Microsoft.Storage
- mi44
- mi45
- mi5
- mi6
- mideploymentinsights

Policy Definitions

Drag a column header and drop it here to group by that column

	Service	Description	Service Resources	Reconciliation State
−	Succeeded	N/A	/subscriptions/a06789e8-93c4-4bed-89c0-40c0be9f2750/resourceGr...	✓ Succeeded

Service: Succeeded
Description: N/A
Service Resources: /subscriptions/a06789e8-93c4-4bed-89c0-40c0be9f2750/resourceGroups/MI43/providers/Microsoft.Storage/storageAccounts/mi43storage1
Reconciliation State: ✓ Succeeded

	Service	Description	Service Resources	Reconciliation State
+	Succeeded	N/A	/services/Azure/ManagedI...	✓ Succeeded

1 | 2 items per page | 1 - 2 of 2 items

How good have you found this content?