Error while Saving Audit and VA Settings

Last updated by | Holger Linke | Apr 28, 2022 at 6:47 AM PDT

Contents

Issue

When saving configuration changes for Auditing or Vulnerability Assessment (VA) settings, the operation might fail with one of the following symptoms:

1. Error message: "Failed to save Auditing settings for server: servername. An unexpected error occured while processing the request. Tracking ID: 'bf66d245-6094-4d96-a270-ef0b8c72cdda'"

2. Set-AzSqlServerAudit fails with error PrincipalNotFound.
   Full error message:
   "Set-AzSqlServerAudit: Failed to add Role Assignment for Storage Account '/subscriptions/<subscription id>/resourceGroups/<resource group>/providers/Microsoft.Storage/storageAccounts/<storage account>'. Response Status Code BadRequest. ResponseContent: {"error":{"code":"PrincipalNotFound","message":"Principal 04cbdf5f02c...421711a does not exist in the directory 11111111-xxxx-yyyy-zzzz-222222222222."}}"

The issue is occurring for attempting it through the Portal and attempting it through PowerShell.
Auditing and Vulnerability assessment can be affected by this issue.

Investigation / Analysis

Cause

The issue is occuring because of using a wrong identifier for a system-assigned identity.

In order to save the auditing or VA results to a storage account behind a VNet firewall, Portal/PowerShell needs to assign the role of "Storage Blob Data Contributor" with the server’s identity to the storage account. The server identity is retrieved and assigned either through its Application ID or its Object ID. For the error to occur, the Portal/PowerShell command is trying to save the Application ID of the Identity Principal instead of the Object ID.

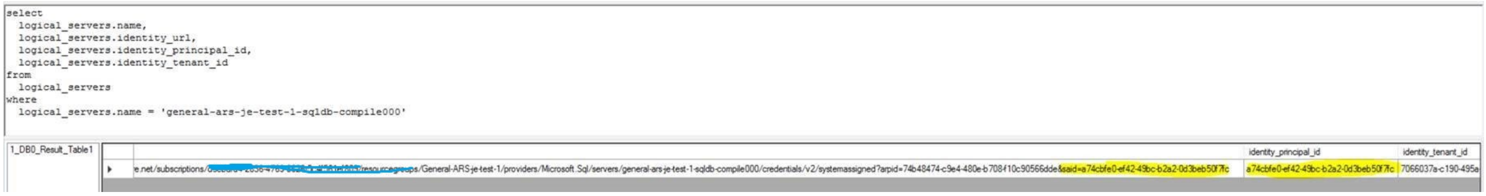
The error "Principal 04cbdf5f02c...421711a does not exist in the directory 11111111-xxxx-yyyy-zzzz-222222222222" is returned when Portal/Powershell is trying to use the Application ID instead of the Object ID in its role assignment.

Troubleshooting steps

You can verify this error conditions on the telemetry by checking the `logical_servers` CMS table in the XTS/HTTP Query Tool. Affected servers have a wrong value in the `identity_prinicipal_id` column. They can be identified by comparing the `identity_url` column with the `said` property on `identity_url` column:

- If `said` and `identity_prinicipal_id` have the same value, then it confirms the cause.
- If `identity_prinicipal_id` is different from the `said` property, or if the affected server does not have a system-assigned identity, then it doesn't match the issue of this article.

```
select name, identity_url, identity_principal_id, identity_tenant_id from logical_servers where name = 'server
```



```
select
  logical_servers.name,
  logical_servers.identity_url,
  logical_servers.identity_principal_id,
  logical_servers.identity_tenant_id
from
  logical_servers
where
  logical_servers.name = 'general-ars-je-test-1-sqldb-compile000'
```

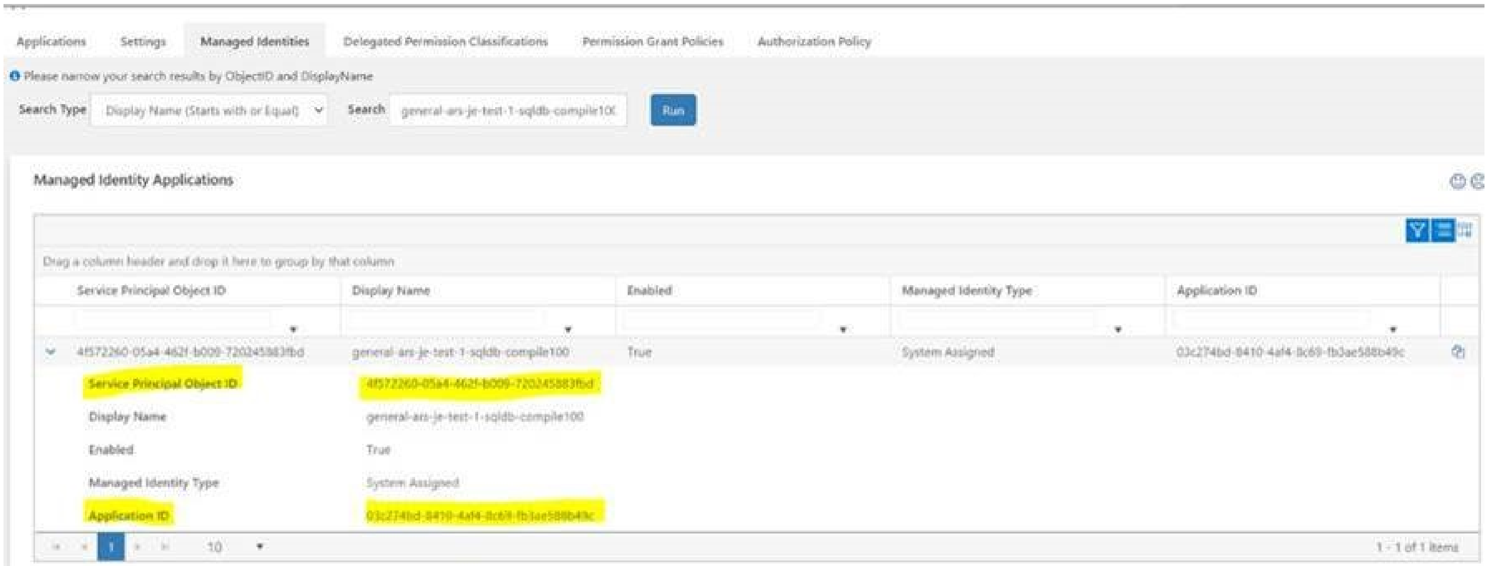
1_DB0_Result_Table1

	identity_principal_id	identity_tenant_id
e.net/subscriptions/[redacted]ps/General-ARS-je-test-1/providers/Microsoft.Sql/servers/general-ars-je-test-1-sqldb-compile000/credentials/v2/systemassigned?arpid=74b48474-c9e4-480e-b708-f10c90566dde&said=a74cbfe0-ef42-49bc-b2a2-0d3beb50f7fc	a74cbfe0-ef42-49bc-b2a2-0d3beb50f7fc	7066037a-c190-495a

```
Sample output for an `identity_url`:

"https://control-westeurope.identity.azure.net/subscriptions/<subscriptionid>/resourcegroups/
<resourcegroup>/providers/Microsoft.Sql/servers/<servername>/credentials/v2/systemassigned?arpid=<GUID1>&said=
-- affected servers have <GUID2> also on `identity_prinicipal_id`
-- note the "systemassigned" type
```

Additionally, in ASC you can see the server's Managed Identity AAD properties - Object ID and Application ID:


Applications
Settings
Managed Identities
Delegated Permission Classifications
Permission Grant Policies
Authorization Policy
Run

Please narrow your search results by ObjectID and DisplayName

Search Type: Display Name (Starts with or Equal)
Search: general-ars-je-test-1-sqldb-compile10C

Managed Identity Applications

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

Service Principal Object ID	Display Name	Enabled	Managed Identity Type	Application ID
4f572260-05a4-462f-b009-720245883fbd	general-ars-je-test-1-sqldb-compile100	True	System Assigned	03c274bd-8410-4af4-8c69-fb3ae588b49c

Service Principal Object ID	4f572260-05a4-462f-b009-720245883fbd
Display Name	general-ars-je-test-1-sqldb-compile100
Enabled	True
Managed Identity Type	System Assigned
Application ID	03c274bd-8410-4af4-8c69-fb3ae588b49c

1 10

1 - 1 of 1 items

Mitigation

See the "More Information" section below for details on the root cause. The reason for the wrong idenity relation has already been fixed, and the current mismatches are left over from before the fix had been applied.

To mitigate, re-assign the Managed Identity to each of the affected servers using PowerShell:

```
Set-AzSqlServer -ResourceGroupName <ResouceGroupName> -ServerName <ServerName> -AssignIdentity
```

This will overwrite the incorrect values with valid settings.

More Information

The following information was collected from the related IcMs and RCAs:

Auditing and Vulnerability assessment use Managed Identity authentication to write to a storage account behind a VNet/firewall. In order to make the auditing saving process easy, we do the following steps in Auditing Portal / PowerShell:

1. Assign Identify on the server if it is not exits.
2. Get server's identity object ID and assign it the role of "Storage Blob Data Contributor" to the storage account.
3. Saving auditing with indication to use Managed Identity authentication

Reference: Audit to storage account behind VNet and firewall

A bug in the server's identity assignment saved the Application ID of the identity instead of the object ID of the identity in the PrincipalId property of server's identity. For example if you run the following PowerShell command, you will see the Application ID instead of the object ID in PrincipalId property:

```
(Get-AzSqlServer -ResourceGroupName <ResourceGroup> -ServerName server_name ).Identity
=> shows Type = SystemAssigned
```

This caused the role assignment in step 2 to fail and raise the error of "Principal does not exist in the directory".

Further details:

> The role assignment on the storage account always happen and when the identity is saved incorrectly, auditing is failing. We made a fix in the portal to handle cases when the Identity is configured incorrectly on the server (the fix doesn't fix the identity itself only the role assignment). However such fix was not applied in the Poweshell commands. If customer wants to use Powershell, they need to fix the servers identity with the mitigation steps provided before.

> This is an issue with the way that SQL has historically identified system-assigned/managed/service identity principals. It's been using the Application ID to identify the principals, instead of the Object ID. One of the values you are comparing is the "source of truth" that SQL recognizes, and the other is the Application ID of the managed identity. These match, but the source of truth should match the Object ID instead.
> This has created a few problems in the past, but it has been creating additional problems as we're trying to transition to using the Object ID, which is the correct value.

Public Doc Reference

[Audit to storage account behind VNet and firewall](#)

Internal Reference

- CSS cases: 2204060050002223, 121011525002066
- https://portal.microsofticm.com/imp/v3/incidents/details/218056010/home
- https://portal.microsofticm.com/imp/v3/incidents/details/300227783/home

How good have you found this content?

🙂 🙁