Restore from LTR backup failing with InternalServerError

Last updated by | Vitor Tomaz | Jun 8, 2022 at 5:31 AM PDT

Contents

Issue

The customer had accidentally deleted an Azure SQL Database. The normal backup retention has already passed, so the restore is attempted from Long-Term Retention Backup.

They are trying the restore from the portal. The LTR backup is available but the actual restore operation is failing with:

```
InternalServerError
An unexpected error occured while processing the request. Tracking ID: '3a82aeaf-86e1-4ef7-ab98-521036ccf5b8'
```

Investigation / Analysis

In a first step, get further details about the failed LTR restore attempt - either in the ASC troubleshooter or in Kusto:

```
// check restore requests -  first filter on server name, than narrow down with request_id
MonManagement
| where TIMESTAMP >= datetime(2021-08-03 14:00:00Z)
| where TIMESTAMP <= datetime(2021-08-03 17:30:00Z)
//| where subscription_id =~ "subscription ID"
//| where logical_server_name =~ "servername"
//| where operation_parameters contains "servername"
| where request_id =~ "CD55FA23-0E0F-4C93-BA27-BC5848ACCEEB"
| project TIMESTAMP, operation_category, operation_type, operation_parameters, originalEventTimestamp, Cluster
| limit 1000
```

In this case, the telemetry showed the following error details:

```
Result: management operation failure
Operation type: Create Restore From Ltr Backup Request
error_code: 0
error_message: The specified target database service level objective or edition is invalid.
```

The customer confirmed that the original database was hosted in an Elastic Pool. There is a documented limitation that for LTR backups, you need to specify the service tier of the target database. The portal however doesn't allow you to specify a target SLO, therefore it cannot restore the LTR backup of an Elastic Pool database.

Mitigation

The solution for this issue is to restore the LTR backup with PowerShell instead of the portal.

```
#Import Az Module
Import-Module Az

# Login with your Azure account
Connect-AzAccount

# Identify available subscriptions
Get-AzSubscription

# Set context to the specific subscription for this task
Set-AzContext -SubscriptionId "<subscription id>"

$rg = "resourcegroupname"
$loc = "westeurope" ## region name
$srv = "servername"
$db = "LTR-Test" ## database name

# List all long term retention backups for a database; use DatabaseState parameter for deleted databases
Get-AzSqlDatabaseLongTermRetentionBackup -Location $loc -ServerName $srv -DatabaseName $db -DatabaseState Dele

## This will return a list of LTR backups, consisting of one or more entries like the following:
## BackupExpirationTime : (some datetime value)
## BackupName           : 6bb07c4f-50ab-469d-93d7-7dd22eb1a3d7;132675912620000000
## BackupTime           : (some datetime value)
## DatabaseName         : LTR-Test
## DatabaseDeletionTime : (some datetime value)
## Location             : westeurope
## ResourceId           : /subscriptions/<subscription_id>/resourceGroups/resourcegroupname/providers/Microsof
##                        ermRetentionDatabases/LTR-Test/longTermRetentionBackups/6bb07c4f-50ab-469d-93d7-7dd2
## ServerName           : servername
## ServerCreateTime     : (some datetime value)
## ResourceGroupName    : resourcegroupname

# get a specific long term retention backup using the backup name
$backup = Get-AzSqlDatabaseLongTermRetentionBackup -Location $loc -ServerName $srv -DatabaseName $db -BackupNa

# display backup details for confirmation
$backup.DatabaseName
$backup.BackupExpirationTime
$backup.BackupTime
$backup.DatabaseDeletionTime

# restore LTR backup as an S3 database - using original name as target
Restore-AzSqlDatabase -FromLongTermRetentionBackup -ResourceId $backup.ResourceId -ResourceGroupName $rg -Serv

# restore LTR backup as an S3 database - using different target name
Restore-AzSqlDatabase -FromLongTermRetentionBackup -ResourceId $backup.ResourceId -ResourceGroupName $rg -Serv


# variation:
# restore a deleted database for which the normal backups still exist (within the backup retention time)
$pit = "2021-07-20 08:00:00" ## point in time of normal backup
$DeletedDatabase = Get-AzSqlDeletedDatabaseBackup -ResourceGroupName $rg -ServerName $srv -DatabaseName $db
Restore-AzSqlDatabase -FromDeletedDatabaseBackup -DeletionDate $DeletedDatabase.DeletionDate -ResourceGroupNam
```

More Information

To monitor the restore progress, you can run the following query on the target server:

```
SELECT major_resource_id
     , operation
     , state_desc
     , percent_complete
     , error_desc
     , start_time
     , last_modify_time
FROM sys.dm_operation_status
WHERE operation = 'CreateRestoreFromLtrBackupRequest'
```

Public Doc Reference

Manage Azure SQL Database long-term backup retention

Get-AzSqlDatabaseLongTermRetentionBackup
Restore-AzSqlDatabase

Get-AzSqlDeletedDatabaseBackup

How good have you found this content?