

# Unable to save auditing due to insufficient storage

Last updated by | Vitor Tomaz | Feb 24, 2023 at 3:28 AM PST

## Contents

- [Issue:](#)
- [How to identify / classify issue:](#)
  - [Using Kusto](#)
- [Workaround or Mitigation:](#)
- [Root Cause for Customer:](#)
- [Reference](#)

## Issue:

Customer is not able to set Server level auditing from Azure Portal, the returned error message is "failed to save auditing settings for server XXXX. An unexpected error occurred while processing the request. Tacking ID: XXXXXXXXXXXXXXXXXXXXXXXX"

## How to identify / classify issue:

### Using Kusto

1. See that the request reached ARM:

```
Execute: [Web] [Desktop] [Web (Lens)] [Desktop (SAW)] https://armprod.kusto.windows.net/ARMProd
HttpIncomingRequests
| where httpMethod == "PUT"
| where targetUri contains "<serverName>" and targetUri contains "auditingSettings"
| where TIMESTAMP >=ago(1d)
```

2. See the status of the in resource provider (RP):

```
Execute: [Web] [Desktop] [Web (Lens)] [Desktop (SAW)] https://sqlazurewus2.kustomfa.windows.net/sqlazure1
MonManagementResourceProvider
| where request_url contains "<serverName>" and request_url contains "auditingSettings"
| where http_verb == "PUT"
| where TIMESTAMP >=ago(1d)
  Or for server level auditing use:
  Execute: [Web] [Desktop] [Web (Lens)] [Desktop (SAW)] https://sqlazureneu2.kustomfa.windows.net/sqlaz
  MonManagementOperations
  | where operation_type == "UpsertServerEngineAuditingPolicy"
  | where subscription_id == "3dbe1a6a-ccdf-4999-8368-81216ab3614a"
  | where operation_parameters contains "servername"
```

Get the request\_id for step 3

### 3. See the logs from the Management Service:

a. Find the region of the server

b. If you haven't found any logs in step 2 and you don't have the request id then run:

```

---
Execute: [Web] [Desktop] [Web (Lens)] [Desktop (SAW)] https://sqlazurewus2.kustomfa.windows.net/sqlazure
MonManagement
| where logical_server_name == "<serverName>"
| where event == "management_workflow_upsert_server_security_engine_auditing_policy_start"
| where TIMESTAMP >=ago(1d)
---
```

Get the request\_id for step 3.c



c. If you found the logs in RP. Take the "request\_id" and run:

```

---
MonManagement
| where request_id == "9DAB381D-2E2F-4411-8BD5-AB253169BDDD"
| where TIMESTAMP >=ago(1d)
---
try to find the error in the event = management_operation_failure
```

In management service, you see below error: Exception of type 'System.Data.SqlClient.SqlException' with Error Number: 1101, State: 12 was encountered. Exception message does not meet compliance 'ObjectName' so it was logged to PiiManagementExceptions with error identifier '6b766caa-7226-4ab1-bc04-bb4dee9c733c'.

### 4. All above logs shows the operation fails, but real exception is not logged. Continue check MonSQLSystemHealth, you can see error 1101 occurred at same time.

```

MonSQLSystemHealth
| where LogicalServerName == 'server_name'
| where message has 'Error: 1101' or message has "Could not allocate"
| project TIMESTAMP, NodeName, error_id, message, LogicalServerName, logical_database_guid
| order by TIMESTAMP desc
```

Sample output:

2020-01-30 09:00:03.74 spid107 Error: 1101, Severity: 17, State: 12. 2020-01-30 09:00:03.74 spid107 [Filtered Args] Could not allocate a new page for database '%1' because of insufficient disk space in filegroup '%2'. Create the necessary space by dropping objects in the filegroup, adding additional files to the filegroup, or setting autogrowth on for existing files in the filegroup.

## Workaround or Mitigation:

Updating SLO or removing some data should resolve that.

## Root Cause for Customer:

Enable server and database auditing settings need to perform operations on the database. Due to insufficient disk space error, saving auditing operation fails.

## Reference

[Incident 312980548](#) 

**How good have you found this content?**

