Custom logging data collected from APIs, Web Apps, Microservices via SeriLog can be sent to the following sinks.

1. Application Insights
2. Azure table storage
3. Azure log analytics
4. Seq Instance on Azure Virtual Machine
5. **Application Insights**

**How**: Nuget package Serilog.Sinks.ApplicationInsights

**Source**: <https://github.com/serilog/serilog-sinks-applicationinsights>

**Asp Core Compatible**: yes

**Configuration**: Get Application insights instrumentation key for the specific API, WebApp, microservice and use the SeriLog Library.

**Pros**: Shows up on application Insights as Custom Events under the serach option

Can filter and search in AI Analytics directly – Seq equivalent

Object de-structuring is supported – can see contents of inserted/updates/deletes

**Cons**: Requires Application insights

Not clear if object de-structuring is supported

Not as intuitive, clear as Seq

1. **Azure table Storage**

**How**: Nuget package Serilog.Sinks.AzureTableStorage

**Source**: <https://github.com/serilog/serilog-sinks-azuretablestorage>

**Asp Core Compatible**: Yes

**Configuration :** Get a storage connection string.WriteTo.AzureTableStorage ("<CloudStorageAccount>").

Go Azure Portal->StorageAccounts->Access keys and copy the connection string

**Demo**: The Api was set up an azure storage account as above.

In log Analytics, under Log Search, telemetry from the Api was received ~~including serilog’s custom with object de-structuring~~. The result was also accessible from the OMS Portal. With further research, dash boards can be created out of custom SeriLog logged logs.

**Pros** : Cheap storage option daily limit and data retention much higher

Can be sent to Analytics as another data Source

Can search/query log in OMS portal

Can set custom alerts on OMS portal

**Cons:** Learning curve\*

Need a way to get storage key from appsettings.json => Cloud Configuration Manager library is not working

1. **Azure Log Analytics**

**How**: Nuget package Serilog.Sinks. AzureAnalytics

**Source**: <https://github.com/saleem-mirza/serilog-sinks-azure-analytics>

**Asp Core Compatible**: yes

**Configuration**: Configure logger by calling WriteTo.AzureLogAnalytics(<workspaceId>, <authenticationId>)

workspaceId: Workspace Id from Azure OMS Portal connected sources.

authenticationId: Primary or Secondary key from Azure OMS Portal connected sources.

**Pros**: Custom Dash Boards, Alerts, Emails etc.

Can aggregate many APIs, Web apps and Microservices in one place (workspace)

**Cons**: Not sure if App services is supported

initial learning curve \*

500MB daily limit, 7 days’ data retention

Suited to VM log data collection

Getting ids from appsettings.json

1. **Seq Instance on Azure Virtual Machine**

**How**: Provision a VM on Azure

**Pros**: Easy searching, filtering, email alerts, charts

**Cons**: VM Potentially costly – utility of Seq equal or less than other options for the cost

Seq requires subscription for SSL

Alerts, emails require separate plugins => scripting the Seq VM instance is more complex