Azure resource logs

07/17/2019 • 7 minutes to read •

In this article

Azure resource logs are platform logs that provide insight into operations that were performed within an Azure resource. The content of resource logs varies by the Azure service and resource type. Resource logs are not collected by default. You must create a diagnostic setting for each Azure resource to send its resource logs to a Log Analytics workspace to use with Azure Monitor Logs, Azure Event Hubs to forward outside of Azure, or to Azure Storage for archiving.

See Create diagnostic settings to send platform logs and metrics to different destinations for details on creating a diagnostic setting and Deploy Azure Monitor at scale using Azure Policy for details on using Azure Policy to automatically create a diagnostic setting for each Azure resource you create.

Send to Log Analytics workspace

Send resource logs to a Log Analytics workspace to enable the features of Azure Monitor Logs which includes the following:

- Correlate resource log data with other monitoring data collected by Azure Monitor.
- Consolidate log entries from multiple Azure resources, subscriptions, and tenants into one location for analysis together.
- Use log queries to perform complex analysis and gain deep insights on log data.
- Use log alerts with complex alerting logic.

Create a diagnostic setting to send resource logs to a Log Analytics workspace. This data is stored in tables as described in Structure of Azure Monitor Logs. The tables used by resource logs depend on what type of collection the resource is using:

- Azure diagnostics - All data written is to the *AzureDiagnostics* table.
- Resource-specific - Data is written to individual table for each category of the resource.

Azure diagnostics mode

In this mode, all data from any diagnostic setting will be collected in the *AzureDiagnostics* table. This is the legacy method used today by most Azure services. Since multiple resource types send data to the same table, its schema is the superset of the schemas of all the different data types being collected.

Consider the following example where diagnostic settings are being collected in the same workspace for the following data types:

- Audit logs of service 1 (having a schema consisting of columns A, B, and C)
- Error logs of service 1 (having a schema consisting of columns D, E, and F)
- Audit logs of service 2 (having a schema consisting of columns G, H, and I)

The AzureDiagnostics table will look as follows:

ResourceProvider	Category	A	B	C	D	E	F	G	H
Microsoft.Service1	AuditLogs	x1	y1	z1					
Microsoft.Service1	ErrorLogs				q1	w1	e1		
Microsoft.Service2	AuditLogs							j1	k1
Microsoft.Service1	ErrorLogs				q2	w2	e2		
Microsoft.Service2	AuditLogs							j3	k3
Microsoft.Service1	AuditLogs	x5	y5	z5					
...									

Resource-specific

In this mode, individual tables in the selected workspace are created for each category selected in the diagnostic setting. This method is recommended since it makes it much easier to work with the data in log queries, provides better discoverability of schemas and their structure, improves performance across both ingestion latency and query times, and the ability to grant RBAC rights on a specific table. All Azure services will eventually migrate to the Resource-Specific mode.

The example above would result in three tables being created:

- Table *Service1AuditLogs* as follows:

Resource Provider	Category	A	B	C
Service1	AuditLogs	x1	y1	z1
Service1	AuditLogs	x5	y5	z5

...

- Table *Service1ErrorLogs* as follows:

Resource Provider	Category	D	E	F
Service1	ErrorLogs	q1	w1	e1
Service1	ErrorLogs	q2	w2	e2

...

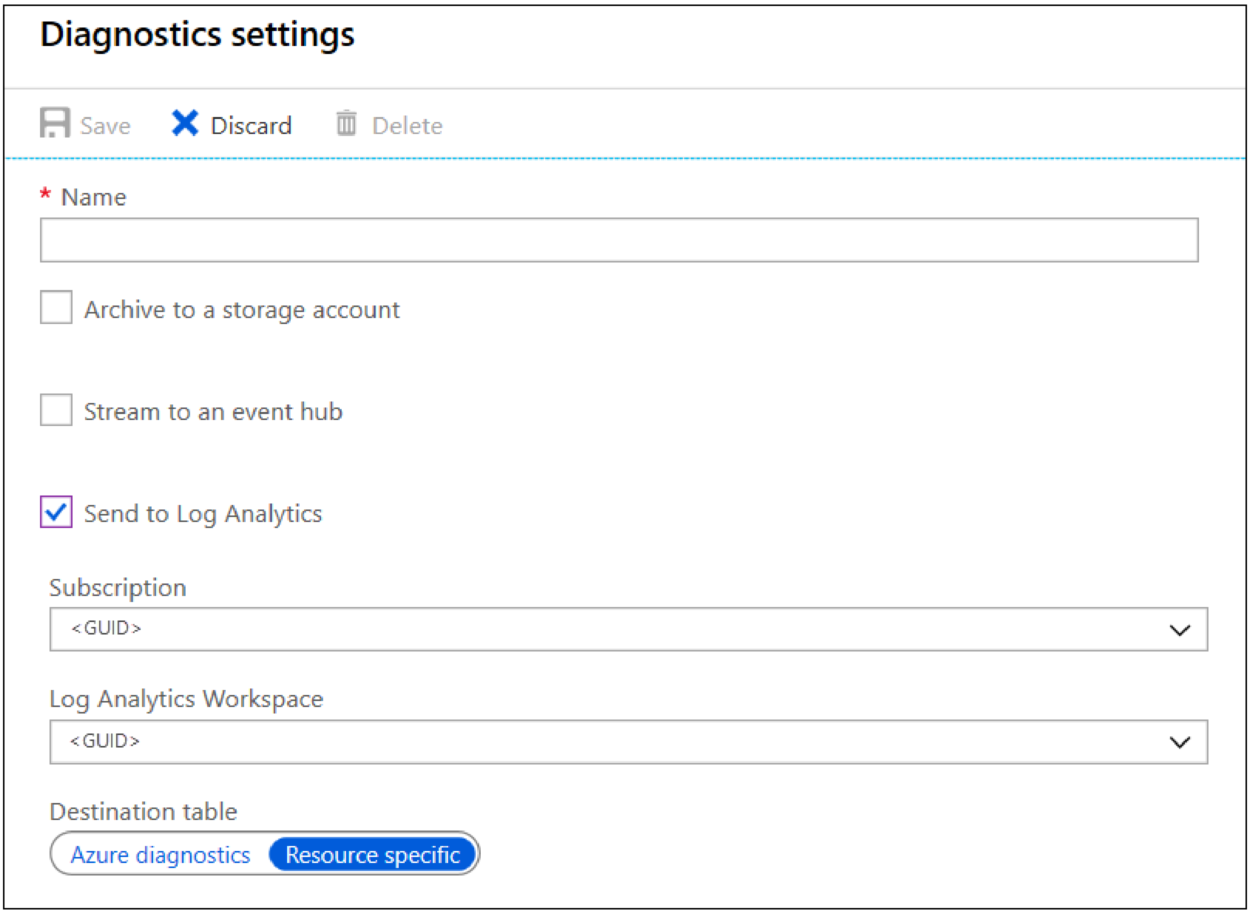
- Table *Service2AuditLogs* as follows:

Resource Provider	Category	G	H	I
Service2	AuditLogs	j1	k1	l1
Service2	AuditLogs	j3	k3	l3

...

Select the collection mode

Most Azure resources will write data to the workspace in either **Azure Diagnostic** or **Resource-Specific mode** without giving you a choice. See the documentation for each service for details on which mode it uses. All Azure services will eventually use Resource-Specific mode. As part of this transition, some resources will allow you to select a mode in the diagnostic setting. Specify resource-specific mode for any new diagnostic settings since this makes the data easier to manage and may help you to avoid complex migrations at a later date.



Diagnostics settings

Save Discard Delete

* Name

□ Archive to a storage account

□ Stream to an event hub

☑ Send to Log Analytics

Subscription

<GUID>

Log Analytics Workspace

<GUID>

Destination table

Azure diagnostics ● Resource specific

> **Note**
>
> For an example setting the collection mode using a resource manager template, see **Resource Manager template samples for diagnostic settings in Azure Monitor**.

You can modify an existing diagnostic setting to resource-specific mode. In this case, data that was already collected will remain in the *AzureDiagnostics* table until it's removed according to your retention setting for the workspace. New data will be collected in the dedicated table. Use the union operator to query data across both tables.

Continue to watch Azure Updates blog for announcements about Azure services supporting Resource-Specific mode.

Column limit in AzureDiagnostics

There is a 500 property limit for any table in Azure Monitor Logs. Once this limit is reached, any rows containing data with any property outside of the first 500 will be dropped at ingestion time. The *AzureDiagnostics* table is in particular susceptible to this limit since it includes properties for all Azure services writing to it.

If you're collecting resource logs from multiple services, *AzureDiagnostics* may exceed this limit, and data will be missed. Until all Azure services support resource-specific mode, you should configure resources to write to multiple workspaces to reduce the possibility of reaching the 500 column limit.

Azure Data Factory

Azure Data Factory, because of a detailed set of logs, is a service that is known to write a large number of columns and potentially cause *AzureDiagnostics* to exceed its limit. For any diagnostic settings configured before the resource-specific mode was enabled, there will be a new column created for every uniquely named user parameter against any activity. More columns will be created because of the verbose nature of activity inputs and outputs.

You should migrate your logs to use the resource-specific mode as soon as possible. If you are unable to do so immediately, an interim alternative is to isolate Azure Data Factory logs into their own workspace to minimize the chance of these logs impacting other log types being collected in your workspaces.

Send to Azure Event Hubs

Send resource logs to an event hub to send them outside of Azure, for example to a third-party SIEM or other log analytics solutions. Resource logs from event hubs are consumed in JSON format with a `records` element containing the records in each payload. The schema depends on the resource type as described in Common and service-specific schema for Azure Resource Logs.

Following is sample output data from Event Hubs for a resource log:

JSON Copy

```json
{
    "records": [
        {
            "time": "2019-07-15T18:00:22.6235064Z",
            "workflowId": "/SUBSCRIPTIONS/00000000-0000-0000-0000-
000000000000/RESOURCEGROUPS/JOHNKEMTEST/PROVIDERS/MICROSOFT.LOGIC/WORKFLOWS/
JOHNKEMTESTLA",
            "resourceId": "/SUBSCRIPTIONS/00000000-0000-0000-0000-
000000000000/RESOURCEGROUPS/JOHNKEMTEST/PROVIDERS/MICROSOFT.LOGIC/WORKFLOWS/
JOHNKEMTESTLA/RUNS/08587330013509921957/ACTIONS/SEND_EMAIL",
            "category": "WorkflowRuntime",
            "level": "Error",
            "operationName":
"Microsoft.Logic/workflows/workflowActionCompleted",
            "properties": {
                "$schema": "2016-04-01-preview",
                "startTime": "2016-07-15T17:58:55.048482Z",
                "endTime": "2016-07-15T18:00:22.4109204Z",
                "status": "Failed",
                "code": "BadGateway",
                "resource": {
                    "subscriptionId": "00000000-0000-0000-0000-
000000000000",
                    "resourceGroupName": "JohnKemTest",
                    "workflowId": "243aac67fe904cf195d4a28297803785",
                    "workflowName": "JohnKemTestLA",
                    "runId": "08587330013509921957",
                    "location": "westus",
                    "actionName": "Send_email"
                },
                "correlation": {
                    "actionTrackingId": "29a9862f-969b-4c70-90c4-
dfbdc814e413",
                    "clientTrackingId": "08587330013509921958"
                }
            }
        },
        {
            "time": "2019-07-15T18:01:15.7532989Z",
            "workflowId": "/SUBSCRIPTIONS/00000000-0000-0000-0000-
000000000000/RESOURCEGROUPS/JOHNKEMTEST/PROVIDERS/MICROSOFT.LOGIC/WORKFLOWS/
JOHNKEMTESTLA",
            "resourceId": "/SUBSCRIPTIONS/00000000-0000-0000-0000-
000000000000/RESOURCEGROUPS/JOHNKEMTEST/PROVIDERS/MICROSOFT.LOGIC/WORKFLOWS/
JOHNKEMTESTLA/RUNS/08587330012106702630/ACTIONS/SEND_EMAIL",
            "category": "WorkflowRuntime",
            "level": "Information",
            "operationName":
"Microsoft.Logic/workflows/workflowActionStarted",
            "properties": {
                "$schema": "2016-04-01-preview",
                "startTime": "2016-07-15T18:01:15.5828115Z",
                "status": "Running",
                "resource": {
                    "subscriptionId": "00000000-0000-0000-0000-
000000000000",
                    "resourceGroupName": "JohnKemTest",
                    "workflowId": "243aac67fe904cf195d4a28297803785",
                    "workflowName": "JohnKemTestLA",
                    "runId": "08587330012106702630",
                    "location": "westus",
                    "actionName": "Send_email"
                },
                "correlation": {
                    "actionTrackingId": "042fb72c-7bd4-439e-89eb-
3cf4409d429e",
                    "clientTrackingId": "08587330012106702632"
                }
            }
            }
        ]
    }
```

Send to Azure Storage

Send resource logs to Azure storage to retain it for archiving. Once you have created the diagnostic setting, a storage container is created in the storage account as soon as an event occurs in one of the enabled log categories. The blobs within the container use the following naming convention:

Copy

```
insights-logs-{log category name}/resourceId=/SUBSCRIPTIONS/{subscription
ID}/RESOURCEGROUPS/{resource group name}/PROVIDERS/{resource provider
name}/{resource type}/{resource name}/y={four-digit numeric year}/m={two-
digit numeric month}/d={two-digit numeric day}/h={two-digit 24-hour clock
hour}/m=00/PT1H.json
```

For example, the blob for a network security group might have a name similar to the following:

Copy

```
insights-logs-
networksecuritygrouprulecounter/resourceId=/SUBSCRIPTIONS/00000000-0000-
0000-0000-
000000000000/RESOURCEGROUPS/TESTRESOURCEGROUP/PROVIDERS/MICROSOFT.NETWORK/NE
TWORKSECURITYGROUP/TESTNSG/y=2016/m=08/d=22/h=18/m=00/PT1H.json
```

Each PT1H.json blob contains a JSON blob of events that occurred within the hour specified in the blob URL (for example, h=12). During the present hour, events are appended to the PT1H.json file as they occur. The minute value (m=00) is always 00, since resource log events are broken into individual blobs per hour.

Within the PT1H.json file, each event is stored with the following format. This will use a common top-level schema but be unique for each Azure service as described in Resource logs schema.

JSON Copy

```json
{"time": "2016-07-01T00:00:37.2040000Z","systemId": "46cdbb41-cb9c-4f3d-
a5b4-1d458d827ff1","category":
"NetworkSecurityGroupRuleCounter","resourceId": "/SUBSCRIPTIONS/s1id1234-
5679-0123-4567-
890123456789/RESOURCEGROUPS/TESTRESOURCEGROUP/PROVIDERS/MICROSOFT.NETWORK/NE
TWORKSECURITYGROUPS/TESTNSG","operationName":
"NetworkSecurityGroupCounters","properties": {"vnetResourceGuid": "
{12345678-9012-3456-7890-123456789012}","subnetPrefix":
"10.3.0.0/24","macAddress": "000123456789","ruleName": "/subscriptions/
s1id1234-5679-0123-4567-
890123456789/resourceGroups/testresourcegroup/providers/Microsoft.Network/ne
tworkSecurityGroups/testnsg/securityRules/default-allow-rdp","direction":
"In","type": "allow","matchedConnections": 1988}}
```

ⓘ **Note**

Platform logs are written to blob storage using **JSON lines**, where each event is a line and the newline character indicates a new event. This format was implemented in November 2018. Prior to this date, logs were written to blob storage as a json array of records as described in **Prepare for format change to Azure Monitor platform logs archived to a storage account**.

Next steps

- Read more about resource logs.
- Create diagnostic settings to send platform logs and metrics to different destinations.

Is this page helpful?

👍 Yes 👎 No