CHAPTER 4

Scenario-Based Examples: Cloud Only

In cloud first architectures, the monitoring perimeters are restricted to components in the cloud. The requirements for capacity, performance, and security monitoring can be met, to a large extent, using solutions available in Azure. The monitoring methodologies and parameters, however, could vary depending on the use cases or scenarios. Depending on whether the environment is greenfield or brownfield, the steps involved in designing the monitoring approach and implementation varies. Hence it is important to have a thorough planning exercise with applied due diligence before we set out to implement the identified solution.

This chapter will provide you a design and implementation reference guide for common cloud-only monitoring scenarios. We will also cover the configuration samples and templates for some of the use cases, including IaaS and PaaS components, which will help you with the implementations.

Design and Implantation Reference

Come of the common architectures for applications using cloud-only services in Azure can be broadly classified as shown in Table 4-1.

Table 4-1.	Architectures	in Our	Scenario
------------	---------------	--------	----------

Architecture	Description
laaS Only	These are "traditional" deployments in Azure, preferred by organizations adopting cloud for the first time. It is also used from on-premises environments that are migrated to cloud using a "Lift and Shift" approach. All application tiers would be front end, middle tier, and databases deployed and managed in VMs, providing customers more control over them.
PaaS Only	This approach is followed by "born in the cloud" organizations. It is also adopted when organizations can claim a certain maturity level in Azure. PaaS offerings, when used at different application tiers, enables customers to focus on the application rather than the nitty-gritty of infrastructure management.
laaS and PaaS	This mix and match approach is commonly used when there is a need for balance between control and flexibility. IaaS services enable organizations to exercise end-to-end control on how they manage the services, while PaaS services offers the flexibility of outsourcing the management to the cloud platform where it deems fit.

Considering the popularity of microservices architecture, we should also add containers to the mix and the monitoring paradigm related to it. The choice of service for hosting containers could be Azure container instances, Kubernetes Services, and associated services like a Azure container registry. The monitoring strategy for a cloud-only environment is done over the following phases – Evaluation, Planning, and Implementation. Let us explore these phases in detail and activities to be covered in each of these phases.

For the ease of understanding the process, let us consider an environment that includes multiple PaaS and IaaS components to be monitored, for example, Virtual machines, webapps, App service environments, AKS, Azure SQl DB, Azure storage, application gateways, Azure load balancers, NSG, etc. We will go through each of these phases, covering the evaluation, design, and implementation processes.

Evaluation

The entry point of designing the monitoring strategy is taking stock of the existing environment in case of Brownfield deployments or evaluating the future state in case of Greenfield deployments. You can use the following pointers to develop a questionnaire for evaluating the environment and to identify the monitoring parameters. Note that the questions are not exhaustive, and you might have to collect more information at the evaluation phase depending on the application architecture.

What are the Azure components needed for monitoring of the application?

For this use case, we have identified the following components – Virtual machines, webapps, App service environment, AKS, Azure SQI DB, Azure storage, application gateway, Azure load balancer, and NSG.

How does the interdependency of components impact monitoring?

Identify dependency of the components being monitored so that false alarms or duplications can be eliminated. For example, a health probe-related error of a load balancer could be a duplicate of a VM health error.

Is it relevant to monitor network connectivity between components or to external endpoints?

When there are many moving parts and dependencies in the application, network connectivity monitoring between components becomes relevant. Network watcher offers extensive network monitoring capabilities using different built-in tools. During the evaluation phase, it is recommended to

identify the tools in the network monitor that can be leveraged to meet the interconnectivity monitoring needs. For example, if your application needs access to an external endpoint, the connection monitor tool can be used to monitor the latency to this endpoint over time.

What components are relevant for monitoring from a business system perspective?

There are multiple metrics available for each of the identified components. Enabling the right metrics and alerts are important to avoid information overload. For example, it makes more sense to monitor metrics of App Service Plans rather than App Service Environment metrics, as App Service plans have a 1:1 mapping to hosts allocated to them.

How do we identity the right metrics for the components?

Now that we have identified the right components to be monitored, the next step is to drill down and finalize the right metrics for monitoring. For example, if you are running a web application in VM with DB hosted in another VM or Azure PaaS services (mySQL, SQL, etc.), it is not relevant to monitor the disk i/o metrics. However, it could be the most important metrics to be monitored in a VM that hosts a database in an IaaS model.

What alerts should be generated and on what is the severity to be assigned?

The goal of monitoring is to flag any anomalies in the system and alert settings are the crucial last mile in this configuration. The specific alert settings such as condition type, threshold, sensitivity, etc., should be considered to assign the right severity to an alert rule.

What are the relevant actions to be taken in response to the generated alerts?

Azure action groups can be used to inform customers about a potential issue through email, SMS, etc. It can also be used to trigger remedial actions through automation runbooks called via webhooks or by invoking logic apps. It is in the evaluation phase that we identity the right course of action and identify the runbooks and logical apps to be created to execute it.

Planning

After the evaluation phase, the next step is to develop the implementation plan. This is not just a technical process but is closely tied to other organizational processes like change management as well. It is one level deeper than the evaluation process, where the parameters of monitoring are locked down and signed off by respective stakeholders. With respect to the different components in the use case, the monitoring configuration would be finalized during the planning phase. A sample outcome of this planning exercise is listed next.

Note: The planning outcome is dependent on the application architecture and the components to be monitored

Components	Monitoring Configuration
Virtual Machines	Performance metrics – CPU, Memory and Disk usage Disk I/O for SQL IaaS VM Guest OS Diagnostic data
App Service	Average response time Data in and Data out Http server errors CPU percentage Memory percentage
Azure SQI DB	Basic metrics Errors QueryStoreRuntimeStatistics QueryStoreWaitStatistics
	(continued)

Components	Monitoring Configuration
Azure Storage	UsedCapacity ResponseType: AuthorizationError ResponseType: ServerBusyError ResponseType: ServerTimeoutError
Application Gateway	Metrics: Failed Requests Metrics: Throughput Metrics: Healthy Host Count Performance log monitoring Firewall log monitoring
Azure Loadbalancer	Data path availability Health probe status
NSG	NSG flow log in Network watcher
AKS	Health status Performance: Node CPU & Memory utilization Container performance monitoring through Azure Monitor for containers
Network Connection	Monitor connection to dependent component IP address through connection monitor of network watcher
Azure Traffic Manager	Queries by endpoint returned Endpoint status by EndPoint

Implementation

The outcome of the planning exercise is a detailed deployment plan for each of those identified monitoring components, which will be executed during the implementation phase. The following, additional pointers should be considered during the implementation phase for a smooth transition to operations.

How can the effectiveness of the implementation plan be tested before a rollout?

It is recommended to have a proof-of-concept period to test the settings and analyze the monitoring data for its effectiveness. This can be leveraged to finetune the configuration and lock down the final settings for implementation.

What roles and responsibilities should be assigned for managing the configurations?

For ongoing maintenance and updating of configurations, the right roles and responsibilities should be identified and assigned during the implementation phase. For example, while the ownership might largely remain with the IT team, application teams would need access to services like log analytics for troubleshooting purposes.

What role-based access control settings should be configured?

Once the roles and responsibilities are finalized, it would give a fair idea of the RBAC permissions to be configured. Any custom roles to be used should also be implemented during this phase

What services and configurations should be used for auto-remediation?

Options like runbooks and logic apps are available to be incorporated with alerts for auto-remediating issues. These runbooks and logic apps will be developed, tested, and deployed during the implementation phase.

What are the plans for operations training and handover?

The operations team should be given adequate training and documentation so that they can take over the monitoring of the components in the cloud. This effort required in the handover depends to a great extent on the expertise of the operations team in handling cloud monitoring.

In the next section we will cover the implementation details more closely for a few sample use cases.

Monitoring a Simple WebApplication on Azure

Let us consider the following use case of a three-tier web application with the sample architecture given in Figure 4-1.



Figure 4-1. Sample three-tier application

As established in the planning section, monitoring will be configured for the following components.

Components	Monitoring Configuration
Virtual Machines	Performance metrics – CPU, Memory and Disk usage Disk I/O for SQL laaS VM Guest OS Diagnostic data
Azure Loadbalancer	Data path availability Health probe status Load balancer health status

Components	Monitoring Configuration
NSG	NSG flow log in Network watcher
Network	Monitor connection to dependent component IP address
connection	through connection monitor of network watcher

Virtual Machines

The parameters to be monitored here are CPU, memory, and disk usage.

CPU Monitoring

CPU monitoring is available through the basic metrics of a VM without any additional configuration. This information can be accessed from VM settings \blacktriangleright Monitoring \triangleright Metrics (Figure 4-2).



Figure 4-2. VM metrics

Memory Percentage

To get memory usage details, you will have to enable VM insights from VM Settings ➤ Monitoring ➤ Insights (preview). Provide the workspace subscription and name and click enabled (Figure 4-3).

* Workspace Subscription 🚯	
	<u>•</u> ••
Choose a Log Analytics Workspace 🚯	
azloganlaytics [eastus]	~
Note: If the virtual machine already has either SCC the Microsoft Monitoring Agent (MMA) extension to the configured work	DM or OMS agent installed locally, will still be installed and connected space.
Enable	

Figure 4-3. Workspace details

Once enabled, you see a notification that insights is being enabled. Note that it could take some time for the monitoring data to be collected and available (Figure 4-4).



Figure 4-4. Notification on enabling Azure Monitor

Once enabled, a confirmation message will be displayed (Figure 4-5).



Figure 4-5. Confirmation message

From the VM insights dashboard, you can now view the Available memory usage and also pin the chart to the Dashboard (Figure 4-6).

,O Search (Ctrl+/)	Kesource Group Monitoring P Azure Monitor P Run Diagnostics C F	Refresh 😳 Provide Feedback
 Disaster recovery Update management 	* 🧏 Performance 🗣 Map 🔯 Health	
Inventory	Available Memory 0	Avg Max 50th 10th 5th Min 🖈
Change tracking		
Configuration management	2.008	~
Policies	2,358	
	133	
Run command	1.408	
Run command	1.4G8 953,7M8	
Run command onitoring insights (preview)	1.69 953748 476548	
Run command onitoring Insights (preview) Alerts	1.63 933.7MB 476.6MB 	06 AM
a Run command tonitoring insights (preview) Alerts if Metrics	1.63 953.7M8 475.6M9 	06 AM

Figure 4-6. Available memory

You can also collect the memory usage metrics by enabling the guest OS diagnostics. To enable Guest-level monitoring, browse to VM \triangleright Diagnostic settings \triangleright "Enable guest-level monitoring" (Figure 4-7).



Figure 4-7. Enable guest-level monitoring

By default, the guest VM performance counters for CPU, Memory, Disk, and Network will be enabled (Figure 4-8).



Figure 4-8. Guest VM performance counters

Now the memory metrics will be available from VM ➤ monitoring ➤ Metrics. Select the metrics namespace as Guest (Classic) and the "Available Bytes" memory counter (Figure 4-9).



Figure 4-9. VM memory metrics

Disk Monitoring

To monitor free disk space in a VM, the "%Free space" counter can be used (Figure 4-10).

Search (Ctrl+,)	~ _	+ New chart 🕐 Refresh 🖉	🕈 Share 🗸 🕲 Feedback 🗸		Last 24 hours (A	utomatic -
······································		Avg \LogicalDisk(_Total)	\% Free Space 🖉			
Configuration management		Le Add matrie for Add She	r 34 Analysalition	IS line d	uart 👽 🔲 New alert rule 🛷 Pin to da	shhoard >
Policies		A surney from the	- of chird strong	No. one o		ISTRUGED -
👰 Run command		FrontendVM2	Guest (classic)	esc AcGEEGATION	~ 0)	
Monitoring						
Insights (preview)		100%				
Alerts		20%				
iii Metrics		80%				
Diagnostic settings		20%				1
Advisor recommendations	10	50%				
🟩 Logs		40%				
Gonnection monitor		30%				<u> </u>
Support + troubleshooting		20%				
🧐 Resource health		0%	05.004	e	CC AM	13.644

Figure 4-10. Free disk space metrics

You can fine-tune the settings if you click on "Configure performance counters" from monitoring > Diagnostics settings of the VM and select only the required counters. For example, for SQL VM in the architecture, you need only Disk I/O-related counters. From the diagnostics settings, click on the performance counters tab. Click on the Custom switch and select the disk i/o related counters and click save (Figure 4-11).

,O Search (Ctrl+/)	« 🔒 Save 🗙	Discard		
 Auto-shutdown Backup 	None Configure the	Basic Custom performance counters to collect, and ho	ow often they should be sampled:	Add
Disaster recovery	PERFORM	IANCE COUNTER	SAMPLE RATE (SECONDS)	UNIT
Update management		Disk(_Total)\% Disk Time	60	Percent
😭 Inventory				
🐁 Change tracking	Logical	Disk(_Total)\% Disk Read Time	60	Percent
Configuration management	Logical	Disk(_Total)\% Disk Write Time	60	Percent
Olicies	Logical	Disk(_Total)\% Idle Time	60	Percent
Run command	Logical	Disk(_Total)\Disk Bytes/sec	60	BytesPerSecond
Monitoring	Logical	Disk(_Total)\Disk Read Bytes/sec	60	BytesPerSecond
Insights (preview)	Logical	Disk(_Total)\Disk Write Bytes/sec	60	BytesPerSecond
🔑 Alerts	✓ \Logical	Disk(_Total)\Disk Transfers/sec	60	BytesPerSecond
Metrics	✓ \Logical	Disk(_Total)\Disk Reads/sec	60	BytesPerSecond
Diagnostic settings	✓ \Logical	Disk(_Total)\Disk Writes/sec	60	BytesPerSecond

Figure 4-11. Custom Switch configuration

In the VM settings \succ metrics, select Guest (classic) from the dropdown and select the disk counter (Figure 4-12). Now you can create the alert based on the threshold to be set for Disk I/O operation.

	: $+$ New chart 🕐 Refresh 🖄 Share \vee 🙂 Feedback \vee	Last 24 hours (Automatic - 1 hour)
Update management	Avg (LogicalDisk(_Total)\Disk Writes/sec &/ Add metric "ty Add litter "G Apoly splitting	rule 📌 Pin to dashboard 🗸 🐳
Change tracking Configuration management Policies Run command	BIOLOGY METRIC BANGPART METRIC Address from O SCUV.NT Guest (classic) \$LogicalDisk_Totalb Avg 0 4580 8%	Ν
nitoring	2,586	$-\Lambda$
Insights (preview) Alerts	38A 258A	\
Metrics	396	
Advisor recommendations	394	
Logs	- <u>3%</u>	

Figure 4-12. Disk counter Metrics

Azure Load Balancer

Monitoring configurations are slightly different for basic and standard load balancers. For basic load balancers, the settings are to be enabled from load balancer ➤ Configure monitoring ➤ Diagnostic settings ➤ Add Diagnostic settings.

In the Diagnostic settings, select the storage account, Event Hub, or log analytics workspace where the diagnostics data should be forwarded to. Since data from all components are being sent to log analytics, the same workspace is selected in this example. The diagnostics data for the load balancer will now be available in Log Analytics for review (Figure 4-13).

, bare	X Discard 🗇 Delete	
* Name		
Lbdiag1		~
Arch	ive to a storage account	
Strea	im to an event hub	
🗸 Senc	to Log Analytics	
Subscrip	btion	
		\sim
Log Apr	Notice Warkspace	
azloga		~
uziogu		-
LOG		
	LoadBalancerAlertEvent	
_		
	LoadBalancerProbeHealthStatus	

Figure 4-13. Load balancer diagnostics configuration

A standard load balancer is the recommended one for all new deployments as it supports features like availability zones, HA ports, HTTPS health probes, as well as the flexibility to add a combination of virtual machines, availability sets, and VMSS in the back-end pools. With respect to diagnostics, standard load balancers support multidimensional metrics for health probe status, byte, and packet counters, outbound connection health, etc.

Data Path Availability

This metrics provides visibility on whether the services are available externally, by testing the datapath from within an Azure region to the back-end Azure virtual machine. It helps to identify Azure infrastructurerelated issues, if any. A traffic matching the front-end rule is automatically generated by the platform to create the metrics. The failure could occur if there are no healthy VMs in the back end or if an infrastructure outage has occurred. The recommended aggregate to be used for these metrics is "Average" (Figure 4-14).

,O Search (Ctrl+/)) «	+ New chart O Refres	sh 😰 Share 🗸 🌐 Feedback 🗸		(Last 2	4 hours (Automatic - 15 minute
Tags		Avg Data Path Avail	ability for azralb2 🧷			
X Diagnose and solve problems		🏷 Add metric 🍾 Ad	d filter 🏾 💘 Apply splitting	🔝 Line chart	🗸 📮 New alert rule	🖈 Pin to dashboard 🗸 🔸
Settings		RESOURCE	METRIC NAMESPACE	NETRIC ADDRESSTON		
Frontend IP configuration		azralb2	Load balancer stan 💙	Deta Path Aveilability 🖌 Avg	~ •)	
Backend pools						
F Health probes		110				
Load balancing rules		90				/
Inbound NAT rules		80				
Properties		70				
a Locks		60				
Locks		40				
Export template		30				
Monitoring		20				
Alerts		10				
		0				

Figure 4-14. Data path availability Metrics

You can also select apply Splitting and choose the Frontend IP address or Frontend port as an additional dimension for monitoring (Figure 4-15)



Figure 4-15. Data path availability Metrics Values

Health Probe Status

This metrics gives an overview of the health probe of the application, based on the instance endpoints configured. This metrics will show downtime if the service is unavailable or if there is any configuration like NSG or firewall blocking access to the endpoints (Figure 4-16).

,○ Search (Ctrl+/) 《	+ New chart () Refresh	Share ∨ (;;) Feedback ∨			Last 2	4 hours (Automatic - 15 minutes
🖉 Tags 🔹	Avg Health Probe Status f	or azralb2 🧷				
X Diagnose and solve problems	🏃 Add metric 🥆 Add filter	Apply splitting		🔄 Line chart 🗸	New alert rule	📌 Pin to dashboard 🗸 🕠
Settings	NISOURCE	METRIC NAMESPACE	METRIC	AGERIGATION		
Frontend IP configuration	(azralb2	Load balancer stan 💙	Health Probe Status	Avg	~ 0)	
Backend pools						
1 Health probes	110					
Load balancing rules	90					
inbound NAT rules						
III Properties	70					
Locks	50					
🖳 Export template	40 30					/
Monitoring	20					
Alerts	10					
M Metrics	0 Sat 20	06 AM	12	м	06 PM	
Support + troubleshooting	Health Probe Status (Jug) acrob2 69.84					

Figure 4-16. Health probe status Metrics

Click on "Apply splitting" and select "Backend IP Address" from the listed options. This is optimal as all other values would be the same for all the endpoints in the back-end pool (Figure 4-17).

,O Search (Ctrl+/)	~	$+$ New chart 🕐 Refresh 🖄 Share \vee	🙂 Fee	edback 🗸
P Tags	*	Avg Health Probe Status for azralb2	0	
K Diagnose and solve problems		🏃 Add metric 🍾 Add filter ⁺⊖ Apply sp	litting	
ettings			6	VALUES
Frontend IP configuration		🗇 azralb2, Health Probe Status, Avg 🔞	(€	Select value(s) V
Backend pools				Backend IP Address
Health probes	L	_110		Backend Port Frontend IP Address
Load balancing rules		90		Frontend Port
Inbound NAT rules		80		Protocol Type
		70		

Figure 4-17. Health probe status Metrics values

As can be seen in the next example, one of the endpoints was down, while the other had higher availability (Figure 4-18).

- Add metric 🍾 Add filter 🍹 Apply splitting	🔛 Line chart 🗸	📮 New alert rule 🛛 🖈 Pin to d	ashboard \lor .
azralb2, Health Probe Status, Avg ()			
110			
_100			
90			
80			
70			
60			
50			
40			
30			
_ 20			
10			
0			
Sat 20 06 AM	12 PM	06 PM	

Figure 4-18. Backend IP address status

NSG

For monitoring NSG flow, Azure Network watcher should be enabled for the region. Search for network watcher from all services in the Azure portal. In the overview tab, expand regions and make sure that service is enabled in the target region where the NSG is created. The next step is to enable the NSG flow logs.

From Network watcher \succ Logs \succ NSG flow logs, filter down to the NSGs where the flow logs need to be enabled. Click on the NSG to open the flow log settings.

Select from between Version 1 and Version 2. Version 1 provides ingress and egress traffic information for packets that have been allowed and denied. Version 2 provides additional information on the flow state starting from when the flow is initiated to continuation and termination of the flow along with the traffic bandwidth details.

Also, select the storage accounts to which the flow logs will be stored. The raw data can be downloaded from the storage account, or it can be analyzed using a traffic analytics solution (Figure 4-19).



Flow logs settings

Figure 4-19. Flow log settings

In the same window you can configure the number of days for which the logs will be stored. You can also enable Traffic analytics to visualize the network activity by leveraging the NSG flow logs. It also helps to identify the traffic flow patterns within Azure and also from the internet, giving valuable insights to optimize performance and capacity.

Note that this configuration needs a log analytics workspace where a network performance monitor solution will be installed. The processing interval can be configured to be either 10 minutes or 1 hour (Figure 4-20).

Retention (days) 🚯			
0			3
Traffic Analytics			
Traffic Analytics Azure resources optimization po	provides rich analytics and vis data. Drill through geo-map, ssibilities.	sualization derived from NSG flow logs and oth easily figure out traffic hotspots and get insigh	er hts into
Learn about all f	eatures		
To use this featur recommend that located. Network We also advise t meta-data is add	re, choose an Log Analytics w t you choose a workspace in t k Performance Monitor solution hat you use the same worksp ded to your flow logs data, to	orkspace. To minimize data egress costs, we the same region your flow logs storage account on will be installed on the workspace. ace for all NSGs as much as possible. Additiona provide enhanced analytics.	t is Il
Traffic Analytics status			
Traffic Analytics processing	j interval 🚯		
Every 10 mins	~		
Log Analytics workspace azloganlaytics			>

Figure 4-20. Traffic Analytics processing interval

After the configured processing interval, Open Network Watcher ➤ Logs ➤ Traffic Analytics to view the NSG flow information (Figure 4-21).

) 8	tefresh	Send us your feedback 📄 FAQ				
ew	analytic	cs of traffic flows across host, subnet and VNet units in <u>Flows</u> Subnet VNet				
Tot	al traffic	Malicious traffic Blocked traffic Frequent conversation	frontendvm2 (10.1.1	.7)		
op :	20 IPs w	ith respect to network traffic flow count	VNet name azmon1-vnet		Subnet name default	
	80		Top talking pair frontendvm2 (10.1.1.7) -	azralb2-VM (16	8.63.129.16)	
	60		Total flows: 89		View units :	Flows 🗸
	40		Inbound	83	6	Outbound
	20		14	69	6	
	0		14	69	6	
		IPs				

Figure 4-21. NSG Flow information

You can drill down to additional details such as Malicious traffic, Blocked traffic, or Frequent conversation from the collected NSG flow information at IP/Subnet/VNet level.

Network Connection

The connection watcher service of network watcher enables connectivity monitoring from a virtual machine to another VM, FQDN, URI, or IP address. This is helpful in monitoring dependent application components and to identify if the network traffic is getting blocked. The connection monitor is also capable of providing potential reasons for a connectivity issue such as DNS resolution issue, custom route-related issue, VM security rules, etc. The data is available over a period of time in terms of minimum, maximum, and average latency observed between the VM and the endpoint.

To monitor a network connection using a connection watcher, browse to network watcher ➤ Monitoring ➤ Connection monitor and Click on +Add.

Give a name to the connection monitor, select the source subscription, source virtual machine, destination virtual machine, and port (Figure 4-22).

Dashboard > Network Watcher - Conner	tion monitor			Add connection monitor	>
Network Watcher - Connection	on monitor				
«	1 144			* Name	
D Search (Ctr1+/)	T Mod			connectionmonitor1	~
📍 Overview	Network Watcher Connection	on Monitor enables you to configure and track conn	nection rear	Source	
Monitoring	tells you why it occurred and Learn more.	d how to fix it.		* Subscription	×
A Topology	Name 🔴	Subscription A	Resource	• Virtual machine	_
S Connection monitor	Filter by name	Microsoft Azure Internal Consump	All resou	Businesstier0	~
Network Performance Monitor				Destination	
Network diagnostic tools	NAME RESOURCE G	ROUP SOURCE IS PORT IS DESTINA	NIN	Select a virtual machine Specify manually	I
IP flow verify	No results.			• Virtual machine 😝	
A Next has				SQLVM1	~
se wext hop				Port o	- 1
Effective security rules				1433	
VPN troubleshoot					_
Se Packet capture				 Advanced settings 	
Connection troubleshoot				Add	

Figure 4-22. Connection monitor configuration

If you click on the connection monitor, the details pane will be displayed on the bottom pane. You can click to view the graph in a new window (Figure 4-23).



Figure 4-23. Connection monitor graph

You can also see the grid view and topology view from the connection monitor (Figure 4-24).

	100 %				
Grid view	Topology view				
Hops					
NAME		IP ADDRESS	STATUS	NEXT HOP IP ADDRESS	RTT FROM SOURCE (MS)
Busines	stier0	10.0.0.5	0	10.1.1.5	
SQLVM	1	10.1.1.5	0	-	-
-	-				
Grid view	Topology view				
		 9			

SQLVM1

10.1.1.5

Figure 4-24. Connection monitor grid view

10.0.0.5

Businesstier0

Monitoring Application That Includes PaaS and IaaS Services

Let us consider the sample architecture of a two-tier web application that uses webapp in the front end, connecting to SQL VMs in the back end (Figure 4-25).



Figure 4-25. Sample two-tier application

As established in the planning section, monitoring will be configured for the following components.

Components	Monitoring Configuration
App service environment	Average response time
	Data in and Data out
	Http server errors
	CPU percentage
	Memory percentage
Azure Storage	UsedCapacity
	ResponseType: AuthorizationError
	ResponseType: ServerBusyError
	ResponseType: ServerTimeoutError
Application Gateway	Metrics: Failed Requests
	Metrics: Throughput
	Metrics: Healthy Host Count
	Performance log monitoring
	Firewall log monitoring

104

Azure Storage

UsedCapacity: Azure storage capacity metrics can be leveraged here to monitor the used capacity. While using standard storage, the value is the sum of capacity consumed by all tables, blobs, and queues in the storage. For premium storage and blob storage, values will be equivalent to the capacity used by all blobs in the storage.

In this example, we are using standard storage. From the Storage setting ➤ Monitoring ➤ Metrics, select metrics "Used Capacity" listed in the drop-down under Capacity (Figure 4-26).

sqlcloudwitnessmo	n - Me	trics					Document
,O Search (Ctrl+/)	~ «	+ New chart 🕐 Refresh 😢	🕈 Share 🗸 🙂 Feedback 🗸			(Last 24 hours (Automatic
an iables	•	Chart Title 🖉					
Queue service		🏷 Add metric 🧤 Add filte	er *e Apply splitting		🔄 Line chart 🗸	New alert rul	le 📌 Pin to dashboard
Queues		(LIFTER HALFCELFT	1175.7		-	
Monitoring		sqlcloudwitnessmon	Account ~	Select metric 🗸 🗸	Select aggregation	•	
👎 Alerts		100		CAPACITY		-	
Metrics				Used capacity			
Advisor recommendations		60		Availability			
Monitoring (classic)		70	Select a metric ab	Egress		we:	
💷 Alerts (classic)		60	-	Ingress			
Metrics (classic)		50	· /e	Success EZE Latency		al	

Figure 4-26. Storage metrics selection

ResponseType: "Authorization error" helps to track any unauthorized access of data in the storage account and is important to monitor from a data security perspective. To monitor the ResponseType, select the metrics type as Transactions (Figure 4-27).

,O Search (Cirl+/)	1 man chart () rentenn E	share V () Peedback V			(Last 2	4 hours (Automatic - 15 min	k/165
Table service	Sum Transactions for sqld	loudwitnessmon 🖉					
Tables	🏃 Add metric 🧤 Add filter	C Apply splitting		🔄 Line chart 🗸	Rew alert rule	📌 Pin to dashboard 🗸	
Queue service	MISOURCE	METRIC NAMISPACE	METRIC	ASSRESATION			
Queues	sqkloudwitnessmon	Account	Transactions	Sum	~ •		
Monitoring	-						
Alerts	100						
fit Metrics	50						
Advisor recommendations	70						
Monitoring (classic)	. 60						
Alerts (classic)	40						
Metrics (classic)	30						
Diagnostic settings (classic)	_ 20				1		
					H A A	AAAA	

Figure 4-27. Storage Transactions metrics

Click on "Add Filter" (Figure 4-28).

Dashboard > Monitor - Sto	orage Accounts (preview) > sqlcloudwitnessmon	- Metrics			
sqlcloudwitnes	smon - Me	trics				
,O Search (Ctrl+/)	*	+ New chart 🕐 Refresh 🖆	Share 🗸 🙂 Feedback 🗸			(
Table service	-	Sum Transactions for sqlc	loudwitnessmon 🧷			
Tables		🏷 Add metric 🦘 Add filter	Apply splitting		🗠 Line chart	🗸 📮 New aler
Queue service		RESOURCE	METRIC NAMESPACE	METRIC	AGGREGATION	
Queues		sqlcloudwitnessmon	Account	Transactions	 ✓ Sum 	~ 0
Monitoring						
💴 Alerts						
iii Metrics						

Figure 4-28. Storage Transactions metrics filter

Select Property as ResponseType and value as "AuthorizationError" (Figure 4-29).

D Search (Cirl+/)	*	+ New chart 🚫 Refresh 🕜 Share 🗸 🐵 Feedback 🗸	(Last 24 hours (Automatic - 15 minutes)
		Sum Transactions where Response type = 'AuthorizationError' 🖉	
ile service		🦕 Add metric 🧤 Add filter 🍓 Apply splitting	e chart 🗸 📮 New alert rule 🛷 Pin to da hooard 🗸 \cdots
Files			
able service		(sqlcloudwitnessmon, Transactions, Sum () (Sqlcloudwitnessmon, Transactions, Sqlcloudwitnessmon, Transactions,	AuthorizationError V
Tables			
Jueue service		_1.10	
Queues			
fonitoring		6.02	
Alerts		6.00	
Metrics		632	
Advisor recommendations		.040	
Aonitoring (classic)		630	
Alerts (classic)		0.10	
Alerts (classic)			

Figure 4-29. "AuthorizationError" filter

App Service

The identified metrics for App Service are the following:

- Average response time: It is the time taken in milliseconds by the app to respond to serve requests.
- Data in and Data out: Monitors the data in and out of the bandwidth consumed by the application in MiB.
- Http server errors: Count of requests that result in an Http status code greater than or equal to 500 but less than 600.
- CPU percentage: This metric will be used at the app service plan level to monitor the CPU usage across all instances of the plan.
- Memory percentage: This metric is also used for the app service plan, to monitor the memory usage across all instances.

The first four metrics, that is, the Average response time, Data in, Data out, and Http server errors can be configured from the app service > monitoring > metrics (Figure 4-30).

,© Search (Ctr1+/)	¢	+ New chart 🕐 Refres	h 🖆 Share 🗸 🙂 Feedb	ack 🗸		Last 2	4 hours (Automatic - 15 minut
Mobile	-	Avg Average Respor	ise Time for ASE-WEB-AP	Pazmon Ø			
Easy tables		😓 Add metric 🧤 Add	i filter 💘 Apply splitting		🔄 Line chart 🗸	New alert rule	🖈 Pin to dashboard 🗸
Easy APIs							
PI		ASE-WEB-APPazer	App Service stan	dar V Average Response	AGGREGATION	~ •)	
API definition							
CORS		4.00)es					
lonitoring		3.50sec					
Alerts		340hes					
í Metrics		2.50sec					A
App Service logs		2.00se					
Log stream		1.50sec					
		1.00sec					
Process explorer		\$00.00ms					
upport + troubleshooting		Diec					
Resource health		Lange Barren T	12 PM	05 PM	1.1/21		06 AM
and the second second second		ASE-WEE-APPEartion	ne (vag)				

Figure 4-30. App Service metrics

For CPU and memory percentage monitoring for all instances, browse to app service plan > Monitoring > Metrics and select the metrics (Figure 4-31).

,O. Search (Ctil+/)	"	+ New chart	C Refresh	Share V	Feedback	~				Last 2	4 hours (Automatic - 15 m	ninutes)
iettings	-	Avg CPU Pe	centage for !	ServicePland	19cc40c2-2c	58-4b23 🧷						
Apps		🄛 Add metri	: 👇 Add filter	👌 Apply sp	litting			🔄 Line ch	art 🗸 📮	New alert rule	📌 Pin to dashboard 🛇	
File system storage			land9cc40c2-2c	App Se	weiwer	VETRIC	taoe 🗸	AUCRECATION	v	0		
Scale up (App Service plan)	11							1.00				
Scale out (App Service plan)		_7%										
Resource explorer		-6%										
Properties		_5%										
Locks		4%										
Export template		3%										
onitoring		.2%										
Alerts		1%										
Metrics		016										
				12 PM		06 FM			3.8.21		05 AM	

Figure 4-31. CPU Percentage metrics

Application Gateway

Both metrics and log files should be monitored for an application gateway. The metrics can be monitored the same way as in other components, that is, from Application gateway \blacktriangleright Monitoring \triangleright Metrics (Figure 4-32).

	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Mearesin Er Site	re V G Feedback V			(Last 2	4 hours (Automatic - 15 mi	tute
Rules	 Avg Healthy 	Host Count for a	izmonsejavaappgw 🧷					
Rewrites	🏃 Add metric	🍾 Add filter 🍓	Apply splitting		🔄 Line chart 🗸	Rew alert rule	📌 Pin to dashboard \sim	•
Health probes	RESOURCE		METRIC NAMESPACE	METRIC	ACGREGATION	_		
Properties	(azmonsej	avaappgw	Application gatewa.	Healthy Host Count	∼ Avg	~ 0)		
Locks								
Export template	0.55							
nitoring	0.45							
Alerts	0.40							-
Metrics	0.30							
Diagnostic settings	0.25							
1.000	0.20							
Logs	0.15							
Backend health	0.10							
Connection troubleshoot	0							
port + troubleshooting		12 PM	05 FR		A.I.21	067	м	

Figure 4-32. Healthy host count metrics

The healthy host count and unhealthy host count information can be further drilled down to a specific back-end pool. This is useful when the application gateway is used for multiple applications and you want to monitor the status of one specific application back-end pool (Figure 4-33).

azmonsejavaappgw - I	Metrics	Documentation
,O Search (Citl+/)	≪ + New chart () Refresh @ Share ∨ () Feedback ∨	Last 24 hours (Automatic - 15 minu
Rules	 Avg Unhealthy Host Count for azmonsejavaappgw by BackendP 	Pool HttpSettings 🖉
Rewrites	🏷 Add metric 🧤 Add filter 🤫 Apply splitting	🔛 Line chart 🗸 📮 New alert rule 🔗 Pin to dashboard 🗸
Health probes		
Properties	🐟 azmonsejavaappgw, Unbealthy Host Count, Avg 🕥 🤇 🕞 Backery	IdPool HttpS V
Locks		
Export template	.1.10	
1.5.0		
lonitoring		
Alerts	0.73	
Metrics	0.60	
Diagnostic settings	0.50	
logs	0.40	
cogs	0.33	
Backend health	0.23	
Connection troubleshoot	0.13	
upport + troubleshooting	12 PM 06 PM	Jul 21 06 AM
P Resource health	arronywa.cothet arronywa.poshet 11 0.85	
New support request		

Figure 4-33. Unhealthy host count metrics

To enable performance and firewall log monitoring, configure the diagnostics settings from Application gateway > Monitoring > diagnostic setting > Add diagnostic setting (Figure 4-34).

Application gateway	agnostic settings			
,O Search (Ctrl+/)	U Refresh			
 ♣ Rules ▲ Rewrites ▲ Note: A state of the s	* Subscription	onsumption (Ocaabf5b-ee7c-4de0-	a0e4-8ab711bd369c) > a	esource group 👩 azmonasejava 🗸 🗸
Properties Locks	Diagnostics settings NAME	STORAGE ACCOUNT	EVENT HUB	LOG ANALY
🖳 Export template	No diagnostic setting	s defined		
Monitoring	+ Add diagnostic sett	ing		
🔍 Alerts	Click 'Add Diagnostic setti	ng' above to configure the collecti	on of the following data:	
Metrics	ApplicationGateway	vAccessLog		
Diagnostic settings	 ApplicationGatewa ApplicationGatewa AllMetrics 	FirewallLog		

Figure 4-34. Add diagnostic settings

Configure the diagnostics information to be sent to a storage account, stream it to Event Hub, or send to a log analytics workspace. In this example, we are sending the Performance log and Firewall log data to the log analytics workspace (Figure 4-35).



Figure 4-35. Application gateway diagnostic settings

Application Insights

Application Insights should be included in the architecture for monitoring of your app service. It can be done using the Application Insights Agent, which is from the Azure portal and ensures a minimum level of monitoring.

Browse to the application \succ settings \succ Application Insights and click on "Turn on Application Insights" (Figure 4-36).



Figure 4-36. Enable Application Insights

Link to an existing application Insights or create a new Application Insights resource (Figure 4-37).

nk to an Application Insights resource	2		
Your app will be connected to an aut Instrumentation key will be added to	o-created Application Insights resource: ASE-WEB-AP App Settings. This will overwrite any instrumentation I	Pkavembyfb4mj2 cey value in web app configuration file	·5.
 Change your resource 			
Change your resource			
 Change your resource Create new resource 			
Change your resource Create new resource New resource name	* Location		
Change your resource Create new resource New resource name ASE-WEB-APPkavembyfb4mj2	Location North Europe		~
Change your resource Create new resource New resource name AsE-WEB-APPkavembyfb4mj2 Select existing resource	* Location North Europe		~
Change your resource Create new resource New resource name AsE-WEB-APPkavembyfb4mj2 Select existing resource Search to find more resources	* Location North Europe		×
Change your resource Create new resource New resource name ASE-WEB-APPKavemby/b4mj2 Select existing resource Select existing resource Select to find more resources Top S relevant resources Belevant resources Construction Constru	* Location North Europe	ral order	×
Change your resource Create new resource New resource name ASE-WEB-APPkawemby/b4mj2 Select existing resource Select existing resources Top 5 relevant resources - Relevance is o	Location North Europe determined by resource group, location, or in alphabet	ical order.	

Figure 4-37. Select/Create Application Insights

Additional settings such as configuration of the collection level, Profiler, SnapShot debugger, etc., can also be configured from this window (Figure 4-38).



Figure 4-38. Application Insights additional settings configuration

Click apply to complete the configuration of Application Insights. You can also manually instrument the code by using Application insights SDK from an IDE like Visual Studio. It is recommended if additional customizations are required where you want to monitor events or dependencies using custom API calls.

Live Metrics from Application Insights is another useful feature that helps you monitor the metrics and performance counters of applications in real time. It is a noninvasive method for live monitoring of your application. Live Metrics needs the latest version of Application Insights SDK and the Microsoft.ApplicationInsights.PerfCounterCollector package to be installed in your webapp. Unlike other metrics that are aggregated over minutes, Live Metrics data (Figure 4-39) is displayed within the duration of 1 second. It is an on-demand monitoring mechanism where the live streaming starts when you open the tab. It does not persist data in storage or log analytics and it is free of charge.



Figure 4-39. Live Metrics Stream

Monitoring Multi-Region WebApplication

Let us consider the sample architecture of a multi-region web application that uses Traffic manager for DNS-based load balancing (Figure 4-40).



Figure 4-40. Multi-region web application sample architecture

As established in the planning section, monitoring will be configured for the following components. Other components like app service plan, storage, etc., are have already been covered in the previous section.

Components	Monitoring Configuration			
Azure SQI DB	Basic metrics			
	QueryStoreRuntimeStatistics QueryStoreWaitStatistics			
Azure Traffic Manager	Queries by endpoint returned Endpoint status by EndPoint			

Azure SQI DB

The basic metrics for SQl DB are available from SQl DB \succ Performance \succ metrics (Figure 4-41).

,O Search (Ctrl+/)	+ New chart 🕐 Refresh	h 🖆 Share 🗸 🙂 Feedback 🗸		
ecurity *	Chart Title 🧷			
Advanced Data Security	🏃 Add metric 🔭 Add	l filter 🍓 Apply splitting	l	🚈 Line chart 🗸 📮 New alert rule
Auditing				
Dynamic Data Masking	azmonsql	Sql database standa Y	Select metric Select	ect aggregation
Transparent data encryption			BASIC	
stelligent Performance	-100		Blocked by Firewall	
Performance overview	80		CPU limit CPU percentage	
Performance recommendati	70	Salart a mateir a	CPU used	
Query Performance Insight	50	Select a metric a	Data IO percentage	
Automatic tuning	50	**/~	Data space allocated	
onitoring	40	Filter + Split 🛛	Plot multiple metrics 🕑	Build custom dashboards
Alerts	_30	Apply filters and splits to identify	Create charts with multiple metri	ics Pin charts to your dashboards
Metrics	20	outlying segments	and resources	
	10			

Figure 4-41. Azure SQL DB Metrics

We will also configure the SQL performance logs to be sent to the Log analytics workspace for centralized monitoring. From the SQL database settings > Monitoring > Diagnostic settings, add the Diagnostics settings and select the identified logs (Figure 4-42).

- Basic metrics: These metrics contain information like DTU/CPU percentage, limit, data read and log write percentage, firewall connections, etc.
- QueryStoreRuntimeStatistics: Information on CPU usage and query duration statistics during query runtime.
- QueryStoreWaitStatistics:Gives insights on what aspects the queries are being waited on, that is, CPU, Locking, log, etc.
- Errors: These metrics give insights on SQL errors that occur in the database.

Diagnostics settings

R Save X Discard 🛍 Delete
Stream to an event hub
Send to Log Analytics
Subscription
Log Analytics Workspace azloganlaytics (eastus) V
LOG
SQLInsights
AutomaticTuning
QueryStoreRuntimeStatistics
QueryStoreWaitStatistics
Errors

Figure 4-42. Select Logs

To view the data and gain additional insights, you can add the solution named Azure SQL Analytics (Preview) in Azure Log analytics (Figure 4-43).

Azure SQL Analytics (Pre adopanloytics	eview)					
🖰 Refresh 🛛 📪 Logs						
Last 24 hours						
TOP METRIC UTILIZATION PER DATA Q DATABASE	BASE METRIC	0 VALLE (%)	DATABASE FLEET OVERVIEW Resources per utilization bucket	0	Query duration in seconds	
🧧 azmonsqlserver.azmonsql	WORKERS	0.5	1		20µ 35 15µ 10µ 5µ	7/21/2019 22:0:00 PM
			6.00 PM 12.00 AM 6.00 AM 12	DO PM	6.00 PM 12.00 AM	600 AM 1200 FM

Figure 4-43. Azure SQL Analytics

Traffic Manager Queries by Endpoint Returned

These metrics can be used to view the number of requests received by a traffic manager over a period of time. This information can also be split across endpoints (Figure 4-44).

Ø Search (Ctrl+Λ)	*	+ New chart O Refresh	🗠 Share 🗸 🙂 Feedback	~		Last 24 hours (Automatic - 15 minut
🖗 Tags		Sum Queries by Endp	oint Returned for azmonde	mo by Endpoint Name 🧷		
K Diagnose and solve problems		🏷 Add metric 🍾 Add	filter *d Apply splitting		🔄 Line chart 🗸	📮 New alert rule 🔗 Pin to dashboard 🗸
ettings		MESOURCE	METRIC NAMISPACE	MITRIC	ASSRELATION	
Configuration		azmondemo	Traffic Manager pro	✓ Queries by Endpoin ✓	Sum	V C Endpoint Name C
Real user measurements						
Traffic view		1,10				
Endpoints		0.90				1
Properties		0.00				
Locks		0.70				
Export template		0.50				
onitoring		0.40				
Alerte	1	0.30				
Alerts		.0.20				
Metrics		0.10				
Diagnostic settings			25.414	12.0	u :	06.814

Figure 4-44. Queries by endpoint-returned metrics

Endpoint Status by Endpoint

This metric is used to show the status of endpoints of a traffic manager. It has two values: 1 if the endpoint is up, and 0 if the endpoint is down (Figure 4-45).

p Search (Ctrl+/)	*	+ New chart O Refresh	n 🕼 Share ∨ 🙂 Feedback י	~		Last 2	4 hours (Automatic - 15 minute
👂 Tags	-	Sum Endpoint Status	by Endpoint for azmondemo	by Endpoint Name 🦉			
K Diagnose and solve problems		🏷 Add metric 🧤 Add	filter "@ Apply splitting		🔄 Line char	t 🗸 📮 New alert rule	🖈 Pin to dashboard 🗸 🔸
ettings		MESOURCE	METRIC NAMESTACE	METRIC	AGGREGATION		
Configuration		azmondemo	Traffic Manager pro	Endpoint Status by	Sum	~ O (4 E	ndpoint Name 🕲
Real user measurements							
Traffic view		_ 80					
Endpoints		_70					1
Properties		60					
Locks		50					
Export template		40					
fonitoring		_ 30					
Alerts	1	- 20					
i Metrics		10					
Diagnostic settings		0				04 DB4	
A		14121	05 AM	123		06 PM	

Figure 4-45. Endpoint status filtered by endpoint Name

Monitoring of Container-Based Microservices

Let us consider a typical AKS deployment architecture as shown next for this use case (Figure 4-46).



Figure 4-46. AKS Architecture

As established in the planning section, monitoring will be configured for the following components.

Components	Monitoring Configuration
AKS	Health status
	Performance: Node CPU and Memory utilization
	Container performance monitoring

Azure offers container monitoring through Azure Monitor for container solutions. It collects performance and memory metrics from nodes as well as containers deployed in those clusters. This information is collated and available for review in log analytics. Container monitoring information, which includes inventory, performance, logs, and events – along with AKS control plane data – is also captured and made available in the Log Analytics workspace. The following information is available from Azure Monitor for containers:

Average processor and memory utilization of containers deployed in AKS clusters;

Identify the location of container and pod to get a better understanding of the container/pod performance;

Monitor cluster behavior under different loads, which helps in capacity planning and future expansion of the cluster;

Configure thresholds and alerts to notify administrators about potential resource bottlenecks. The functionality of Azure Monitor for containers is depicted in the next diagram (Figure 4-47).



Figure 4-47. Azure Monitor for containers

To enable Azure Monitor for containers in your AKS cluster, browse to AKS cluster > Monitoring > Insights. Select the log analytics cluster to which you want to link the cluster and click enable (Figure 4-48).

,O Search (Ctrl+/)	Azure Monitor
Node pools (preview)	
Upgrade	With Azure Kubernetes Service, you will get CPU and memory usage metrics for each node. In addition, you can enable container m
🗹 Scale	get insights into the performance and health of your entire Kubernetes cluster.
Networking	You will be billed based on the amount of data ingested and your data retention settings. We create a default workspace for you if subscription this cluster is in. It will take 5-10 minutes for the onboarding process to complete.
A Dev Spaces	Lase more shout container health and performance monitoring [2]
6 Deployment center (preview)	Learn more about container reality and performance monitoring Es
Policies (preview)	Learn more about pricing 🖄
Properties	Log Analytics workspace o
Locks	loganlyask 🗸
関 Export template	Enable
Monitoring	Namegane Service Node Tenencys
🝷 Insights	(d) • (d) • (d) • (d) •

Figure 4-48. Select Log Analytics workspace

Once enabled, you can see a segregated view of information from Cluster, Nodes, Controllers, and containers in the cluster (Figure 4-49).

,O Search (Ctrl+/)	*	🖰 Refresh 📔 🍣 Vi	ew All Cl	usters	[0]	Monitor re	source gro	up	🖸 Fe	edback	~		
Node pools (preview)	^	Time range = Last 6 ho	urs) (* ¥ Ad	dd Filte						View V	Vorkbool	s v
O Upgrade		Cluster Nodes	Co	ntrolle	ers	Contair	ners			For	ums 🔼	Learn m	ore 🔼
Scale					-								
Networking		Search by name			Metric	Memory	working se	• •	Min	Avg 5	oth 90th	95th	Max
🖧 Dev Spaces													19 items
Deployment center (preview)		NAME S	TA 9	_↓ 95	тн і	POD	NODE	RES	UPTI	TREND 95T	H % (1 BAR	= 15M)	
Policies (preview)		🔲 omsagent	0	18% 9	91.7	omsage	aks-nod	0	22 h				
Properties		omsagent	0	18% 1	107	omsage	aks-nod	0	22 h				
Locks		ineapster-na	0	13% 1	11.8	heapste	aks-nod	0	22 h				
Export template		coredns	0	8% 1	13.8	coredns	aks-nod	0	22 h				
Monitoring		tomcat-cntnr	0	8% 3	362,	tomcat	aks-nod	0	22 h				
💡 Insights		coredns	0	8% 1	13.4	coredns	aks-nod	0	22 h				
Metrics (preview)		beanster	0	896 1	15.2	beanste	aks-pod	0	22 h				
💭 Logs		incapster		070	19.5	neapste	015-1100	•	22 11				
Support + troubleshooting		azure-ip-ma	v	2% 5	5.42	azure-ip	aks-nod	0	22 h				
New support request		i main	0	2% 1	10.4	kuberne	aks-nod	0	22 h				~

Figure 4-49. Azure Monitor for containers sample output

If you want to do a deeper analysis using container logs, browse to the log analytics workspace that the AKS cluster was linked to. Browse to general ➤ Logs and view the logs from "ContainerLog" (Figure 4-50).

loganlyask - Logs								¢ ×
D Search (Ctil+/)	≪ I New Query 1* +					Help	Settings	Ξ S.
Cverview	loganlyask	茸	D Run (Time rang	ge: Last 24 hours	Save	👁 Copy	🔁 Export	+ Ne
 Activity log Access control (IAM) 	Schema Filter	«	ContainerLog limit 50					
🖉 Tags	Filter by name or type	,0°						
X Diagnose and solve problems	1	I [™] Collapse all						
Settings	Active							
Locks	 Ø loganlyask Containertosiohts 	\$						
🐯 Export template	• 🗉 Containerimagelnv	ventory	Completed. Showing results	from the last 24 hours.				00
Advanced settings	• 🗏 Container/inventory	y	TABLE ALICHART COL	umns ~				
	ContainerLog	۲	Drag a column header and drop i	it here to group by that column				
General Quick Start	ContainerNodelinv	og	TimeGenerated (UTC) > 2019-07-22103/24:10.050	Computer aks-nodepool1-45400822-0	TimeOfCommand [UTC 2019-07-22103/26/07/0	∇ 0 30 4	ContainerID 10148244bc3325	26433130
Workspace summary	 Insignaments KubeEvents 		> 2019-07-22T03:24:12.699	aks-nodepool1-45400822-0	2019-07-22103:25:07.0	00 4	10148244bc3325	28455/54
View Designer	• 🖾 KubeNodeInventor	ry .	> 2019-07-22T03:24:13.141	aks-nodepool1-45400822-0	2019-07-22103:26:07.0	00 4	10148244bc3325	28455854
🗐 Logs	KubePodInventory KubeSarvices		> 2019-07-22T03:24:13.641	aks-nodepool1-45400822-0	2019-07-22103:26:07.0	10 4	40148244bc3325	28455/50
🤣 Solutions			> 2010_07.22703.2413.648	ake-oodaooodt.45400899.0	2019-07-22103 26:07 0	m	01/182/16/2220	70,155/5,1

Figure 4-50. View Container log

Additional metrics information is available by default from Monitoring \blacktriangleright Metrics (preview) (Figure 4-51).

Search (Cirl+/)	+ New chart 🕐 Refresh 🗔 Diagnose 🗸 😢 Share 🗸 🛛	I Feedback ∨	(Last 24 hours (Automatic - 1 minute)
Node pools (preview)	Sum Number of pods in Ready state for aksaadburns1	0	
Upgrade	😓 Add metric 🧤 Add filter 🍓 Apply splitting	🔛 Line chart 🗸 📮 New i	alert rule 🔗 Pin to dashboard 🗸 🚥
Scale	REDOURCE METRIC HAMIESPACE	METER ADDRESS	
Networking	aksaadburns1 Container service (💙	Number of pods in V Sum V	
Dev Spaces		Number of pods by phase	
Deployment center (preview)	- 20	Number of pods in Keady state	
Policies (preview)		Total number of available cpu cores in a managed	
Properties	14		
Locks			
Export template			
nitoring			
nitoring		1	
itoring Insights Metrics (preview)			

Figure 4-51. Additional container metrics

You can view the following information from the Figure 4-51: Number of pods by phase, Number of pods in the ready state, Total amount of memory available in a managed cluster, and Total number of available CPU cores in a managed cluster.

You can even drill down to further details by applying dimensions. For example, for the metrics "Number of pods in ready state," you can drill down to the number of pods in a specific namespace (Figure 4-52).



Figure 4-52. Number of pods in ready state

Action Groups and Alerts

Configuration of metrics and logs is useful only if you are able to generate alerts in case of any anomalies. Action groups are used to define the notification mechanism for alerts. Action groups consist of the following:

- Name: Unique name for the action group;
- Action type: Action group can be configured to notify stakeholders through emails, SMS, push notifications, or voice messages. It can also call a logic app, Function, Webhook, Automation runbook, or generate an ITSM ticket;

• **Details:** These are configurations specific to the Action type selected.

Action groups will be called during an alert configuration, thereby triggering a notification when an alert condition is met.

To configure Action groups, open Azure Monitor \succ Alerts \succ Manage actions \succ Add action group.

Provide information such as Action group name, Short name to be included in notifications, and the subscription and resource group where the action group will be created. In this, for example, we will trigger an automation runbook using the action group, and hence "Automation Runbook" is selected as the Action type. Click on Edit details to configure the runbook to be used (Figure 4-53).

	Commence de la commence de la				
Action group name ()	azmondemorule				
Short name 0	azmondemo				
Subscription O					
Resource group 🛛	Default-ActivityLogAlerts (t	o be created)			
Resource group 0	Default-ActivityLogAlerts (t	o be created)			
Resource group 🛛	Default-ActivityLogAlerts (t	ACTION TYPE	STATUS	DETAILS	,
Resource group () ctions ACTION NAME Demonotification	Default-ActivityLogAlerts (t	ACTION TYPE	status V	DETAILS Édit details	
Resource group () ctions ACTION NAME Demonotification Piease configure the action	Default-ActivityLogAlerts (t	ACTION TYPE	STATUS	DETALS Edit details	

Figure 4-53. Select Action Type

In the Configure runbook tab, select the Runbook source. It can be built in runbook or runbooks created by users. There are built-in runbooks available to perform the following activities: Stop, Restart, Scale up, Scale down, and Remove VM. In this example, we have selected the Scale up VM option. Select a subscription and an automation account in the subscription. All other configurations can be default. Click on Ok. It will take you back to the action group configuration window. Click Ok there to create the action group (Figure 4-54).

Configure Runbook
* Run runbook Enabled Disabled
* Runbook source 1 Built-in User
* Runbook Scale Up VM 🗸 🗸
This runbook will set the Azure virtual machine to the next larger size within the size group.
* Subscription
* Automation account omsrgautmn

Figure 4-54. Select Automation Runbook

Now that an action group is created, let us see how to use the action group in alerts. To create an alert for a VM using collected metrics, browse to the VM \blacktriangleright Monitoring \triangleright Alerts and Click on the new alert rule. Click on Add under Condition and click on the metrics to be used. In this example, let us create an alert rule based on the CPU percentage (Figure 4-55).

Create ru Rules managem	ule Mark	Choose a signal below and configure the logic on the next screen to de	fine th	e alert condition.		
[]	* RESOURCE	Signal type 🛛	Moni All	itor service		v
ليو	FrontEndVm1	Displaying 1 - 20 signals out of total 47 signals				
	Select	SIGNAL NAME	SIGN	AL TYPE	MONITOR SERVICE	1.
		Percentage CPU	\sim	Metric	Platform	
۲¢	+ CONDITION	Network In Billable	\sim	Metric	Platform	
Ľ	No condition defined, click on 'Add condition' to	Network Out Billable	\sim	Metric	Platform	
	Add	Disk Read Bytes	~	Metric	Platform	

Figure 4-55. Select signal logic for alert rule

Configure the alert condition. Here the action group will be generated when the CPU utilization is greater than 80% for 5 minutes (Figure 4-56).

0%						
3 AM Percentage CPU (Avg) frontendvm1 2.27 %	4 AM	5 AM	6 AM	7 AM	8 AM	
Alert logic Threshold 🕢	ic	Dj	mamic			
Operator 🕢 Greater than	~	* Aggregation typ Average	ж О	* Thresho	ld value 🜒	√ %
Condition preview Whenever the percentag Evaluated based on	e cpu is greater than &	10 percent				
* Aggregation granulari	ity (Period) 🚯		Frequency of	of evaluation 🕤		~

Figure 4-56. Alert configuration

The next step is to configure an action in the Action group. Click on Add and select the Action group that we created earlier. Essentially this alert setting scales up the VM to the next available size when the CPU percentage is higher than 80% for 5 minutes (Figure 4-57).

reate n	ala		
ules manager	NOTE.		
F	* RESOURCE	For metric and log alerts, action groups selected must be from subscriptions other than the alert rule's subscription Subscription ()	in the alert rule's subscription. For activity log alerts, action groups can be sele
노		Microsoft deare internal Consumption (Read-55) ee Re-4deb	uðo4-ðuls?+Hol360ot
	FrontEndVm1	P search to falter items	
	Select	ACTION GROUP NAME	CONTAIN ACTIONS
		Application Insights Smart Detection	2 Email Azure Resource Manager Role(s)
Ê	+ CONDITION	Application Insights Smart Detection	2 Email Azure Resource Manager Role(s)
\sim	SWhenever the Percentage CPU is Greater t	Application Insights Smart Detection	2 Email Azure Resource Manager Role(s)
		azmondemorule	1 Automation Runbook
	Add	iassoms_AG_1	1 Email, 1 Webhook
		iaasoms_AG_2	1 Email, 1 Webhook
ġ	ACTIONS	iaasoms_AG_3	1 Email
1	ivo conjugurea actions	iaasoms_AG_4	1 Email, 2 Webhook(s)
	Add	language AC A	1 Forell, 1 Michigan

Figure 4-57. Select action group

Note You can add additional action groups to the same rule, for example, to send an email to the IT team so that when the configured condition matches, the IT team is alerted while the VM is also scaled up automatically.

The next step is to set an alert rule name, description, and severity; and then click on "Create Alert rule: to complete the configuration (Figure 4-58).

ALERT DETAILS	
* Alert rule name 🚯	
CPU utilization	1
Description	
CPU utilization	ג
	1
* Severity 💿	
Enable rule upon creation	
Yes No	
It can take up to 10 minutes for a metric alert rule to become active.	

Figure 4-58. Create Alert rule

Summary

In this chapter, we have reviewed the different phases of implementation of monitoring for cloud-only scenarios: that is, evaluation, planning, and implementation. We also covered the different possible scenarios and implementation details of respective components. Azure Monitoring enables customers to configure alerts based on the monitoring configurations: that is, notifying stakeholders or taking actions such as executing runbooks. The configuration details of the same was also covered in this chapter.