Level: Advanced

Microsoft Azure Exam AZ-104 Certification

← Back to the **Course**

Azure Monitoring - **Practice Mode**

Completed on Sat, 18 Oct 2025

1st	2/5	40.00%	FAIL
Attempt	Marks Obtained	Your Score	Result

Download Report

Domain wise Quiz Performance Report

No.	Domain	Total Question	Correct	Incorrect	Unattempted	Marked for Review
1	Monitor and maintain Azure resources	5	2	3	0	0
Total	All Domains	5	2	3	0	0

Review the Answers

Filter By

Question 1 — Correct

Domain: Monitor and maintain Azure resources

An organization wants to optimize the cost of its Azure resources by analyzing usage patterns. An Azure administrator decides to interpret Network Out Total and Network In Total metrics for their Virtual Machine Scale Set (VMSS) over the past 30 days using the Azure Monitor Metrics workspace. The administrator uses the default aggregation method, "Average," to identify patterns in network traffic over time.

Proposed Solution: The administrator should switch to the "Total" aggregation method instead of "Average" to evaluate the cumulative network usage of the VMSS effectively. Is this proposed solution correct? (Select Yes or No)

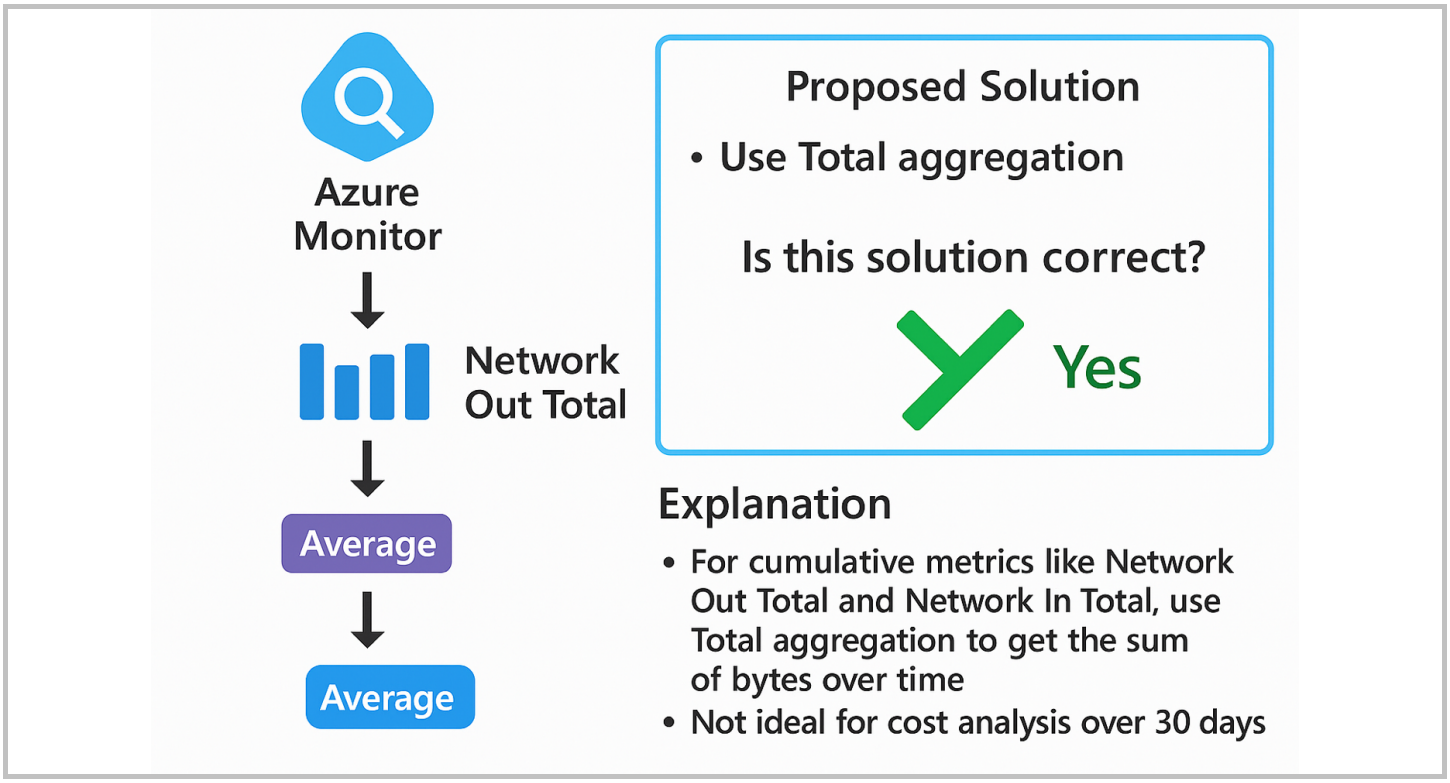
◯ A. Yes right

B. No

Explanation:

Correct Answer: A

The proposed solution is valid and correct. When the goal is to analyze usage patterns for cost optimization, the "Total" aggregation method is more suitable for cumulative metrics like "Network Out Total" and "Network In Total."



The "Average" aggregation method, while useful for understanding general trends and patterns, does not provide an accurate picture of the total data transfer volume over a long period. Using "Average" can underestimate resource consumption by averaging out traffic spikes and dips.

By contrast, the "Total" aggregation method sums up the metric values over the selected time range, providing a precise measurement of the cumulative data usage. This allows the administrator to accurately identify peak usage periods, determine whether the current configuration is cost-effective, and make informed decisions about network bandwidth or resource scaling adjustments to reduce costs.

Reference:

Azure Monitor metrics aggregation and display explained

Ask our Experts

Did you like this **Question?**

Question 2 Incorrect

Domain: Monitor and maintain Azure resources

View Case Study

You need to ensure that logs from all production subscriptions (Prod-A, Prod-B, and Prod-C) are centralized in a single Log Analytics workspace named CentralWorkspace. Logs must have a 365-day retention policy, and the setup should minimize redundant data transfer costs. Which of the following step should you perform?

A. Enable the diagnostic settings for each subscription and direct all logs to CentralWorkspace

B. Configure data collection rules (DCR) in each subscription and route logs to CentralWorkspace wrong

C. Use Azure Policy to enforce the diagnostic settings across all subscriptions and connect them to CentralWorkspace right

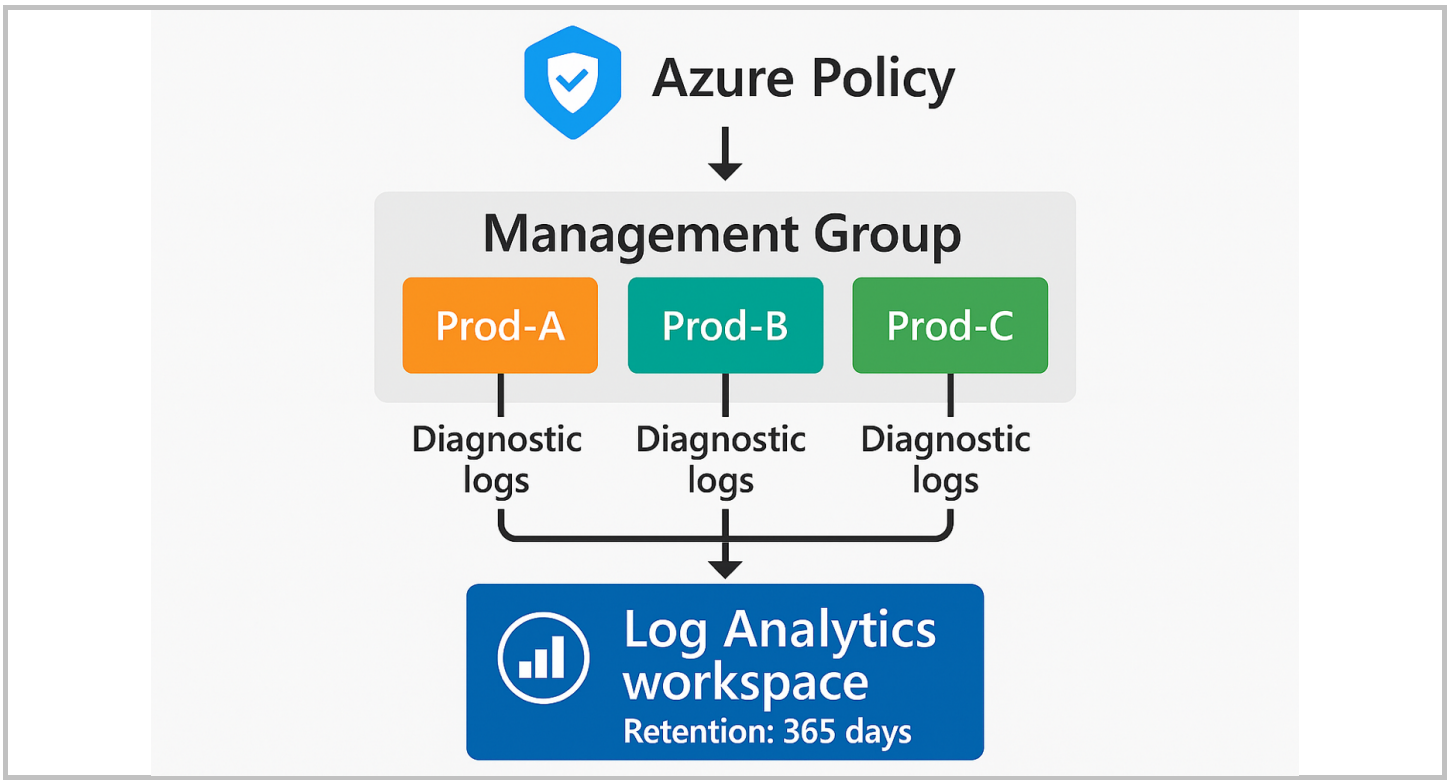
D. Set up cross-resource query capabilities across existing Log Analytics workspaces instead of centralizing

---

Explanation:

Correct Answer: C

Option C: Use Azure Policy to enforce the diagnostic settings across all subscriptions and connect them to CentralWorkspace is correct because Azure Policy is the most effective way to enforce diagnostic log settings across multiple subscriptions. By defining a policy that applies to all the subscriptions (Prod-A, Prod-B, and Prod-C), Contoso can automate the process of ensuring that all logs are sent to a single centralized Log Analytics workspace, named CentralWorkspace. This approach removes the need to manually configure each resource or subscription. Once the policy is in place, it will automatically align any non-compliant resources by ensuring they direct logs to the appropriate workspace. Applying the policy at the management group level ensures that it automatically applies to all associated subscriptions. This not only helps streamline log management but also reduces the risk of unnecessary data transfers, thus lowering costs. Additionally, the retention period of 365 days can be set in the Log Analytics workspace to meet regulatory and compliance standards. Overall, using Azure Policy to manage log settings improves operational efficiency while ensuring consistent adherence to best practices across the entire organization.



Option A: Enable the diagnostic settings for each subscription and direct all logs to CentralWorkspace is incorrect because while enabling diagnostic settings and directing logs to CentralWorkspace could centralize logs, this option requires configuring settings individually for each resource or subscription. This manual approach is error-prone, inefficient for a large-scale environment, and fails to enforce consistent logging policies. Additionally, it does not ensure compliance with the requirement to minimize redundant data transfer costs, as manual configurations could lead to duplications or misconfigurations.

Option B: Configure data collection rules (DCR) in each subscription and route logs to CentralWorkspace is incorrect because Data Collection Rules (DCR) are primarily used to configure and manage telemetry collection for Azure Monitor metrics and custom log data. While they offer flexibility for data ingestion, they are not the recommended approach for enforcing diagnostic settings at a subscription level. Moreover, DCRs are more suitable for custom scenarios, not for the centralized, automated enforcement needed across multiple subscriptions as described in the case study.

Option D: Set up cross-resource query capabilities across existing Log Analytics workspaces instead of centralizing is incorrect because cross-resource queries allow you to analyze data from multiple Log Analytics workspaces without centralizing logs into a single workspace. While this can be useful for querying distributed data, it does not address the requirement for centralized log collection and a unified retention policy. Additionally, querying across multiple workspaces can result in increased latency and complexity, making it unsuitable for scenarios requiring streamlined, centralized monitoring and compliance.

Reference:

https://learn.microsoft.com/en-us/azure/azure-monitor/essentials/diagnostic-settings-policy

Ask our Experts

Did you like this **Question?**

Question 3 Incorrect

Domain: Monitor and maintain Azure resources

View Case Study

A network administrator suspects that intermittent connectivity issues between on-premises and Azure VNETs are caused by an IP configuration mismatch on one of the VPN tunnels. Which KQL query should you use in Log Analytics to identify the issue?

A. AzureDiagnostics | where ResourceType == "VirtualNetworkGateway" | where Message contains "IPMismatch" | project TimeGenerated, ResourceId, Message right

B. NetworkConnections | where Protocol == "TCP" | where Direction == "Inbound" and Port == 443 | summarize count() by RemoteIP, ResourceId wrong

C. AzureNetworkLogs | where Category == "GatewayDiagnosticLogs" | where Message contains "Latency" | summarize AvgLatency = avg(ResponseTime) by ResourceId

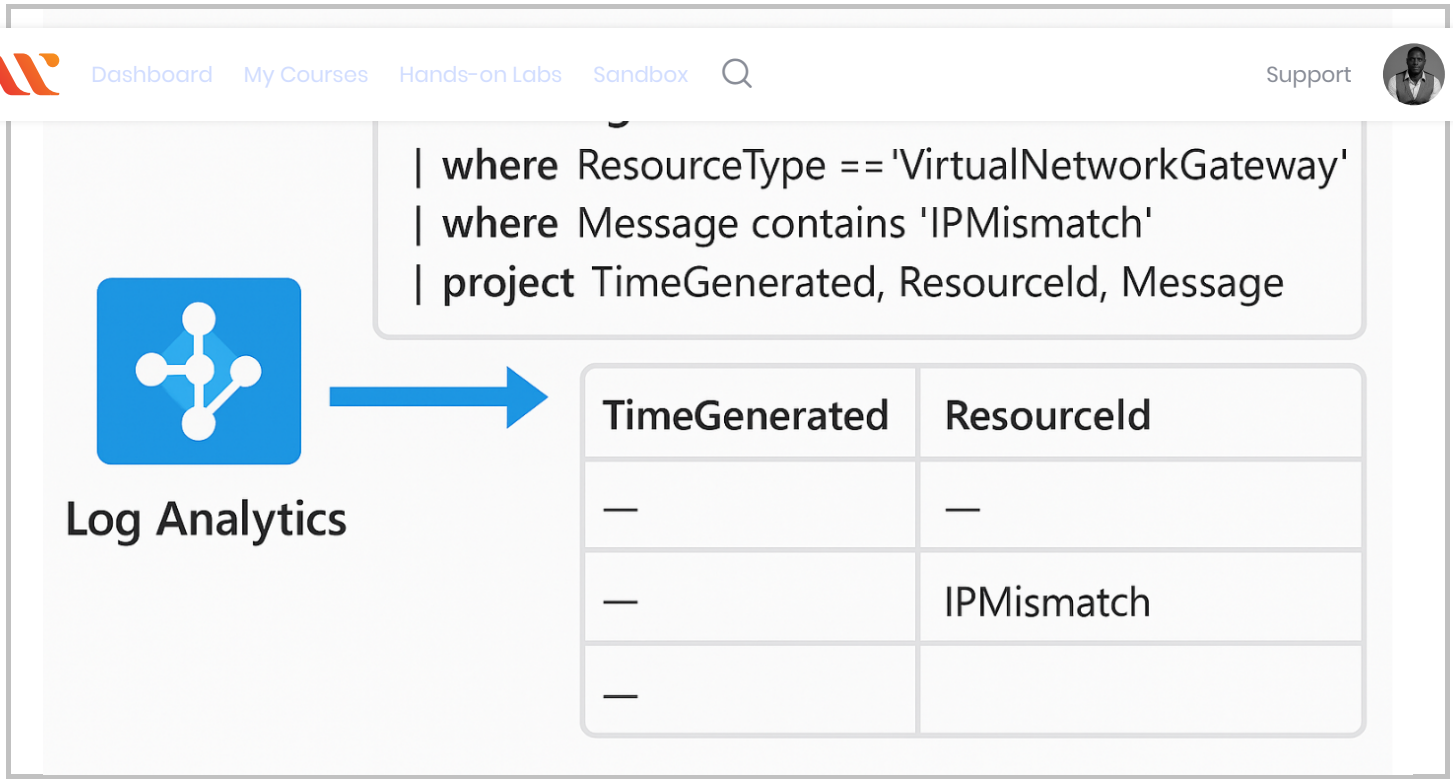
D. VMConnection | where Status == "Failed" | summarize FailureCount = count() by ResourceId

---

Explanation:

Correct Answer: A

Option A is correct because the query is designed to detect potential IP configuration problems within a Virtual Network Gateway. It specifically looks at data in the AzureDiagnostics table, which contains logs for various Azure resources, including network-related events. By filtering for ResourceType = "VirtualNetworkGateway", the query focuses only on logs related to the Virtual Network Gateway, ensuring that the analysis remains relevant without being overwhelmed by data from other resources. The query also searches for the term "IPMismatch" within the log messages, which helps identify entries related to IP configuration issues that could be causing intermittent connectivity problems. Additionally, the use of the project operator retrieves only the essential fields TimeGenerated, ResourceId, and Message making the results more concise and easier to interpret. Overall, this query effectively targets the specific problem described in the scenario, offering a streamlined approach to identifying and troubleshooting the issue.



Option B is incorrect because this query inspects TCP connections to resources, filtering for inbound connections on port 443 (typically HTTPS). While this might help analyze general network activity, it does not focus on Virtual Network Gateways or IP mismatches, which are the primary concern here. Additionally, it summarizes connection counts, which does not provide detailed insights into intermittent connectivity issues caused by configuration mismatches. This query is better suited for analyzing connection trends or traffic patterns.

Option C is incorrect because this query analyzes logs for latency-related metrics using the AzureNetworkLogs table and the GatewayDiagnosticLogs category. While it provides information about response times and latency for gateways, it does not address IP mismatches. Focusing on latency metrics would be irrelevant in this context because the issue is likely caused by configuration errors, not performance bottlenecks.

Option D is incorrect because this query investigates failed connections to virtual machines by analyzing the VMConnection table. Although connection failures could indicate potential issues, the scenario is specific to a Virtual Network Gateway and involves an IP mismatch, not VM connectivity. This query is unrelated to the described problem and would not yield any actionable insights for the network administrator.

References:

https://learn.microsoft.com/en-us/azure/azure-monitor/reference/tables/azurediagnostics

https://learn.microsoft.com/en-us/azure/azure-monitor/logs/log-query-overview

https://learn.microsoft.com/en-us/azure/azure-monitor/essentials/diagnostic-settings

Ask our Experts

Did you like this **Question?**

Question 4 Correct

Domain: Monitor and maintain Azure resources

View Case Study

Contoso needs to create an alert rule that triggers when the average CPU usage of any production VM exceeds 85% for 5 minutes. Notifications should be sent to both the DevOps team via email and the Operations team via SMS. What is the most efficient configuration to achieve this?

A. Create an alert rule scoped to the VM resource group, define a CPU usage condition, and assign an action group with email and SMS contacts

B. Create separate alert rules for each VM, define a CPU threshold condition, and link them to individual action groups for email and SMS notifications

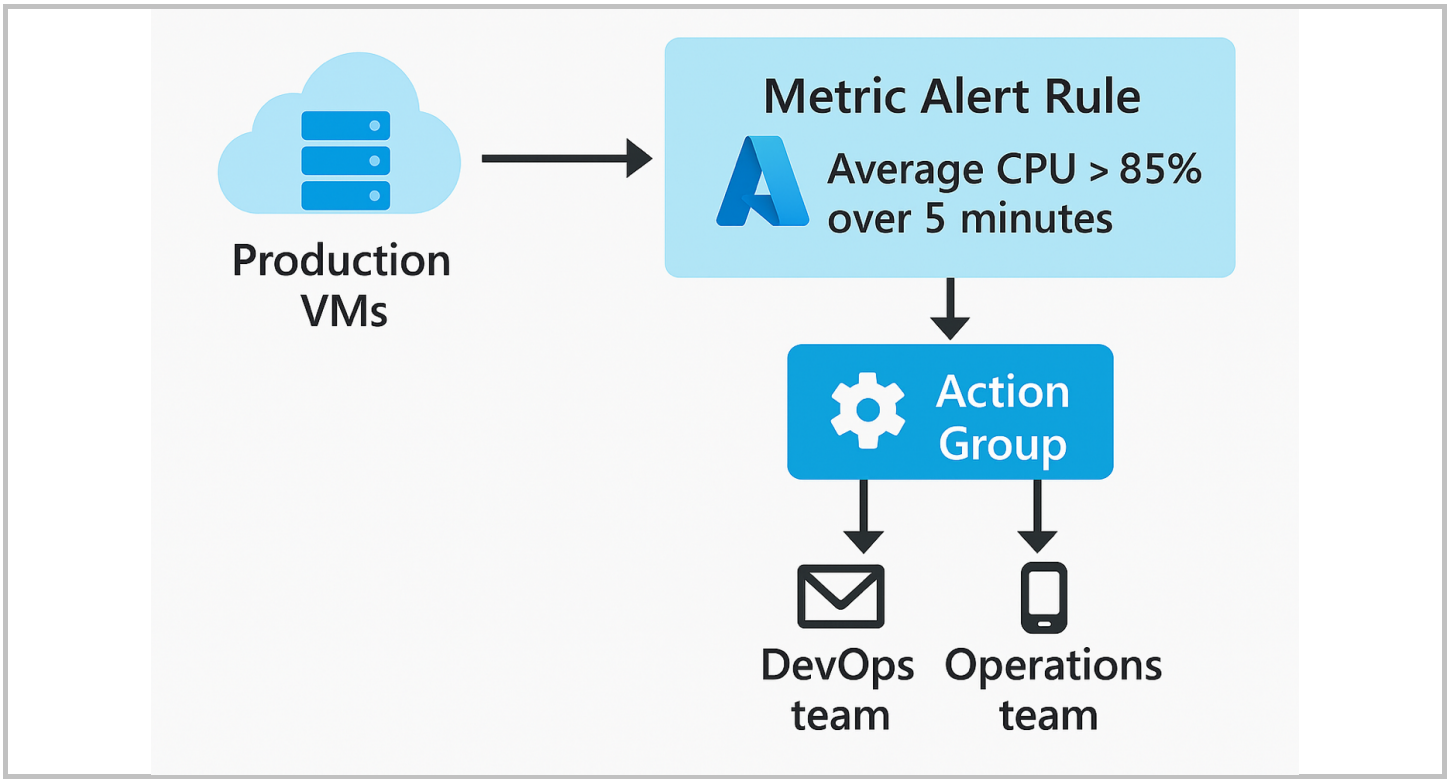
C. Configure a metric alert rule at the subscription level, define the CPU usage threshold, and link it to a unified action group with email and SMS contacts right

D. Use Log Analytics to query CPU metrics and set up a log alert with integrated action groups for email and SMS notifications

Explanation:

Correct Answer: C

Option C: Configure a metric alert rule at the subscription level, define the CPU usage threshold, and link it to a unified action group with email and SMS contacts is correct because this is the most efficient and scalable solution. Metric alert rules at the subscription level enable monitoring of all VMs across the subscription and automatically include new VMs that meet the defined criteria (e.g., tagging or resource type). By defining a threshold for average CPU usage (e.g., exceeding 85% for 5 minutes), the rule ensures proactive monitoring. Linking the alert rule to a unified action group ensures that notifications are sent simultaneously via email to the DevOps team and via SMS to the Operations team. Automatically includes all current and future VMs under the subscription. Reduces complexity compared to configuring multiple rules or scoping to a resource group. Using a single action group simplifies notification configuration and ensures that all recipients (DevOps and Operations teams) are informed simultaneously. Metric alerts are optimized for performance and scalability compared to log-based alerts, especially for high-frequency signals like CPU usage.



Option A: Create an alert rule scoped to the VM resource group, define a CPU usage condition, and assign an action group with email and SMS contacts is incorrect because while creating an alert rule scoped to the resource group is possible, it is less efficient for scenarios involving multiple VMs in production. Resource group-level scoping is limited because it does not dynamically apply to all VMs if new resources are added to the group after the alert is configured. Moreover, resource group scoping is less precise compared to subscription-level metric alert rules that can inherently apply to all relevant VMs.

Option B: Create separate alert rules for each VM, define a CPU threshold condition, and link them to individual action groups for email and SMS notifications is incorrect because creating separate alert rules for each VM is an inefficient approach, especially in environments with multiple or dynamically scaled VMs. This method introduces high administrative overhead and makes the management of alerts more complex. Additionally, it increases the risk of inconsistency in configuration across VMs. Subscription-level alert rules combined with unified action groups are more scalable and easier to maintain.

Option D: **Use Log Analytics to query CPU metrics and set up a log alert with integrated action groups for email and SMS notifications is incorrect** because log-based alerts can provide detailed and flexible analysis, but they are less efficient than metric-based alerts for monitoring resource-specific metrics like CPU usage. Log alerts rely on custom queries and may incur additional latency due to the ingestion of data into the Log Analytics workspace. For real-time scenarios like CPU monitoring, metric alerts are more suitable as they directly pull data from the Azure Monitor metrics pipeline, providing faster and more reliable triggering.

References:

https://learn.microsoft.com/en-us/azure/azure-monitor/alerts/alerts-create-metric-alert-rule

https://learn.microsoft.com/en-us/azure/azure-monitor/alerts/action-groups

https://learn.microsoft.com/en-us/azure/azure-monitor/alerts/alerts-overview

Ask our Experts

Did you like this **Question?**

Question 5 Incorrect

Domain: Monitor and maintain Azure resources

View Case Study

You need to monitor end-to-end connectivity between your on-premises network and Azure resources. Contoso's IT team has configured Connection Monitor in Azure Network Watcher. However, they need to test latency across VPN tunnels proactively. Which additional configuration should you implement to meet this requirement?

A. Add a network topology map to visualize VPN latency metrics in real-time using Network Watcher

B. Set up Traffic Analytics in Azure Network Watcher to analyze real-time data flow and detect latency issues

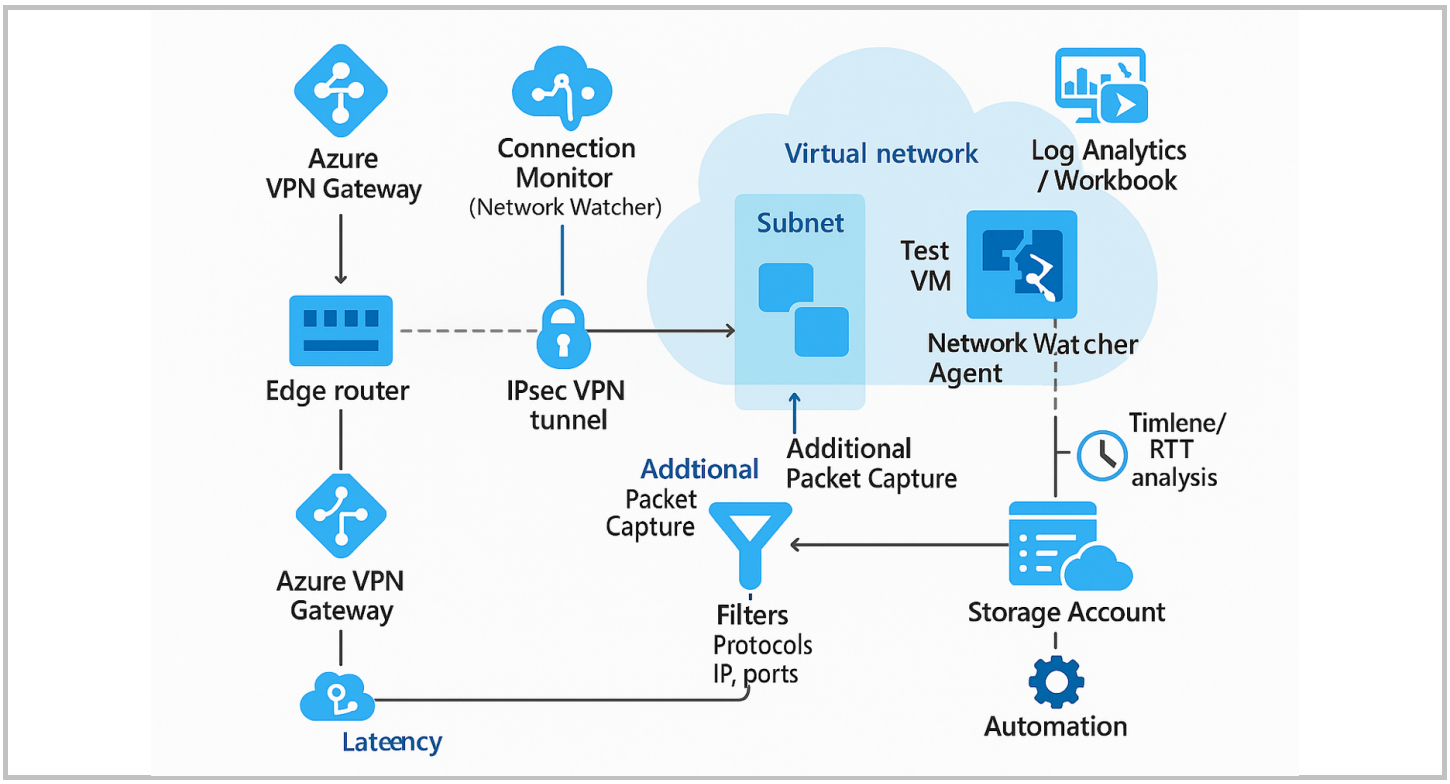
C. Use the IP Flow Verify tool in Azure Network Watcher to validate the IP path and measure latency wrong

D. Enable packet capture in Azure Network Watcher and analyze captured packets for latency patterns right

Explanation:

Correct Answer: D

Option D: Enable packet capture in Azure Network Watcher and analyze captured packets for latency patterns is correct because Packet capture allows for a detailed analysis of network traffic. By capturing packets as they traverse the VPN tunnels, you can examine timestamps and response delays within the captured data to identify specific latency issues. This method is highly effective for troubleshooting complex network problems where granular, packet-level details are necessary.



Option A: Add a network topology map to visualize VPN latency metrics in real-time using Network Watcher is incorrect because Network topology map is a visualization tool that shows the connections between network resources. It does not provide real-time performance metrics like latency and cannot be used for proactive testing.

Option B: Set up Traffic Analytics in Azure Network Watcher to analyze real-time data flow and detect latency issues is incorrect because Traffic Analytics provides a high-level overview of network traffic patterns and flow. It does not offer the granular, packet-level details needed to measure specific latency across a VPN tunnel.

Option C: Use the IP Flow Verify tool in Azure Network Watcher to validate the IP path and measure latency is incorrect because IP Flow Verify tool is used to check if an IP packet is allowed to or from a specific source/destination and to diagnose connectivity failures. It does not measure or monitor performance metrics such as latency.

References:

https://learn.microsoft.com/en-us/azure/network-watcher/packet-capture-overview

https://learn.microsoft.com/en-us/azure/network-watcher/packet-capture-inspect

https://learn.microsoft.com/en-us/azure/network-watcher/connection-troubleshoot-overview

https://learn.microsoft.com/en-us/azure/network-watcher/network-watcher-overview

Ask our Experts

Did you like this **Question?**

Finish Review