

FCSS_SOC_AN-7.4 Dumps

FCSS - Security Operations 7.4 Analyst

https://www.certleader.com/FCSS_SOC_AN-7.4-dumps.html



NEW QUESTION 1

Which role does a threat hunter play within a SOC?

- A. investigate and respond to a reported security incident
- B. Collect evidence and determine the impact of a suspected attack
- C. Search for hidden threats inside a network which may have eluded detection
- D. Monitor network logs to identify anomalous behavior

Answer: C

Explanation:

Role of a Threat Hunter:

A threat hunter proactively searches for cyber threats that have evaded traditional security defenses. This role is crucial in identifying sophisticated and stealthy adversaries that bypass automated detection systems.

Key Responsibilities:

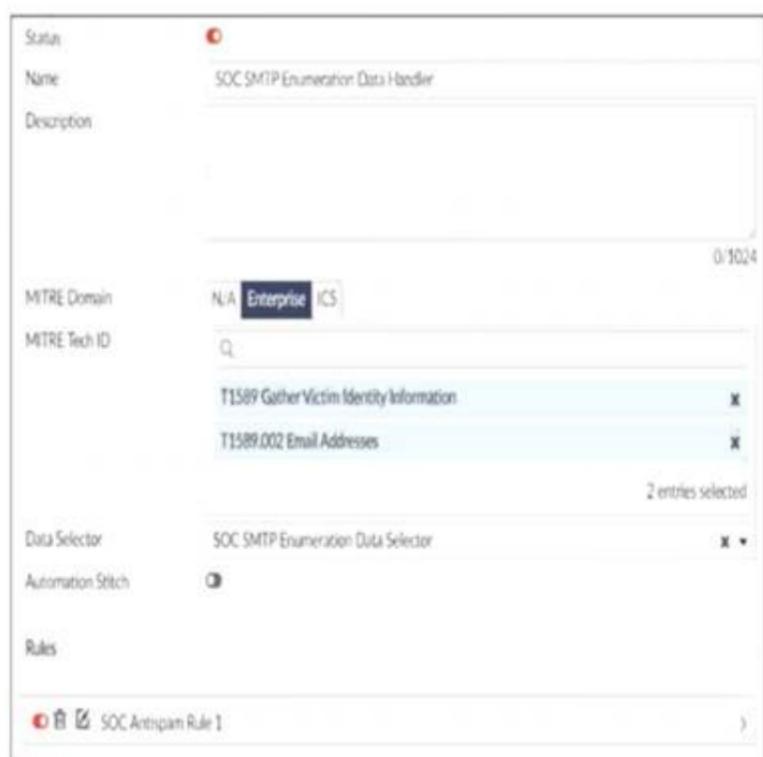
Proactive Threat Identification:

Threat hunters use advanced tools and techniques to identify hidden threats within the network. This includes analyzing anomalies, investigating unusual behaviors, and utilizing threat intelligence.

NEW QUESTION 2

Refer to the exhibits.

Event Handler



You configured a custom event handler and an associated rule to generate events whenever FortiMail detects spam emails. However, you notice that the event handler is generating events for both spam emails and clean emails.

Which change must you make in the rule so that it detects only spam emails?

- A. In the Log Type field, select Anti-Spam Log (spam)
- B. Disable the rule to use the filter in the data selector to create the event.
- C. In the Trigger an event when field, select Within a group, the log field Spam Name (snane) has 2 or more unique values.

Answer: A

Explanation:

Understanding the Custom Event Handler Configuration:

The event handler is set up to generate events based on specific log data.

The goal is to generate events specifically for spam emails detected by FortiMail.

Analyzing the Issue:

The event handler is currently generating events for both spam emails and clean emails.

This indicates that the rule's filtering criteria are not correctly distinguishing between spam and non-spam emails.

Evaluating the Options:

Option A: Selecting the "Anti-Spam Log (spam)" in the Log Type field will ensure that only logs related to spam emails are considered. This is the most straightforward and accurate way to filter for spam emails.

Option B: Typing type==spam in the Log filter by Text field might help filter the logs, but it is not as direct and reliable as selecting the correct log type.

Option C: Disabling the rule to use the filter in the data selector to create the event does not address the issue of filtering for spam logs specifically.

Option D: Selecting "Within a group, the log field Spam Name (snane) has 2 or more unique values" is not directly relevant to filtering spam logs and could lead to incorrect filtering criteria.

Conclusion:

The correct change to make in the rule is to select "Anti-Spam Log (spam)" in the Log Type field.

This ensures that the event handler only generates events for spam emails.

References:

Fortinet Documentation on Event Handlers and Log Types.

Best Practices for Configuring FortiMail Anti-Spam Settings.

NEW QUESTION 3

When configuring a FortiAnalyzer to act as a collector device, which two steps must you perform?(Choose two.)

- A. Enable log compression.
- B. Configure log forwarding to a FortiAnalyzer in analyzer mode.
- C. Configure the data policy to focus on archiving.
- D. Configure Fabric authorization on the connecting interface.

Answer: BD

Explanation:

Understanding FortiAnalyzer Roles:

FortiAnalyzer can operate in two primary modes: collector mode and analyzer mode.

Collector Mode: Gathers logs from various devices and forwards them to another FortiAnalyzer operating in analyzer mode for detailed analysis.

Analyzer Mode: Provides detailed log analysis, reporting, and incident management.

Steps to Configure FortiAnalyzer as a Collector Device:

* A. Enable Log Compression:

While enabling log compression can help save storage space, it is not a mandatory step specifically required for configuring FortiAnalyzer in collector mode. Not selected as it is optional and not directly related to the collector configuration process.

B. Configure Log Forwarding to a FortiAnalyzer in Analyzer Mode:

Essential for ensuring that logs collected by the collector FortiAnalyzer are sent to the analyzer FortiAnalyzer for detailed processing.

Selected as it is a critical step in configuring a FortiAnalyzer as a collector device.

Step 1: Access the FortiAnalyzer interface and navigate to log forwarding settings.

Step 2: Configure log forwarding by specifying the IP address and necessary credentials of the FortiAnalyzer in analyzer mode.

NEW QUESTION 4

Refer to the exhibits.

Playbook status

Job ID	Playbook	Trigger	Start Time	End Time	Status
2024-03-20 08:32:14 770575-07	DOS attack	event:202403201008	2024-03-20 08:32:15-0700	2024-03-20 08:32:15-0700	Failed

Playbook tasks

Task ID	Task	Start Time	End Time	Status
placeholder_8fab0102_0955_447f_872d_220	Attach_Data_To_Incident	2024-03-20 08:32:18-0700	2024-03-20 08:32:18	upstream_failed
placeholder_fa2a573c_ba4f_4565_ba10_4255d	Get Events	2024-03-20 08:32:17-0700	2024-03-20 08:32:18	success
placeholder_3db75c0a_1765_4479_8118_2e1	Create SMTP Enumeration incident	2024-03-20 08:32:17-0700	2024-03-20 08:32:18	failed

Raw Logs

```
[2024-03-20T08:32:18.089-0700] {taskinstance.py:1937} ERROR - Task failed with exception
Traceback (most recent call last):
  File "/drive0/private/airflow/plugins/incident_operator.py", line 218, in execute
    self.epid = int(self.epid)
ValueError: invalid literal for int() with base 10: '10.200.200.100'
```

The DOS attack playbook is configured to create an incident when an event handler generates a denial-of-service (DoS) attack event. Why did the DOS attack playbook fail to execute?

- A. The Create SMTP Enumeration incident task is expecting an integer value but is receiving the incorrect data type
- B. The Get Events task is configured to execute in the incorrect order.
- C. The Attach_Data_To_Incident task failed.
- D. The Attach_Data_To_Incident task is expecting an integer value but is receiving the incorrect data type.

Answer: A

Explanation:

Understanding the Playbook and its Components:

The exhibit shows the status of a playbook named "DOS attack" and its associated tasks.

The playbook is designed to execute a series of tasks upon detecting a DoS attack event.

Analysis of Playbook Tasks:

Attach_Data_To_Incident:Task ID placeholder_8fab0102, status is "upstream_failed," meaning it did not execute properly due to a previous task's failure.

Get Events:Task ID placeholder_fa2a573c, status is "success."

Create SMTP Enumeration incident:Task ID placeholder_3db75c0a, status is "failed."

Reviewing Raw Logs:

The error log shows aValueError: invalid literal for int() with base 10: '10.200.200.100'.

This error indicates that the task attempted to convert a string (the IP address '10.200.200.100') to an integer, which is not possible.

Identifying the Source of the Error:

The error occurs in the file "incident_operator.py," specifically in theexecute method.

This suggests that the task "Create SMTP Enumeration incident" is the one causing the issue because it failed to process the data type correctly.

Conclusion:

The failure of the playbook is due to the "Create SMTP Enumeration incident" task receiving a string value (an IP address) when it expects an integer value. This mismatch in data types leads to the error.

References:

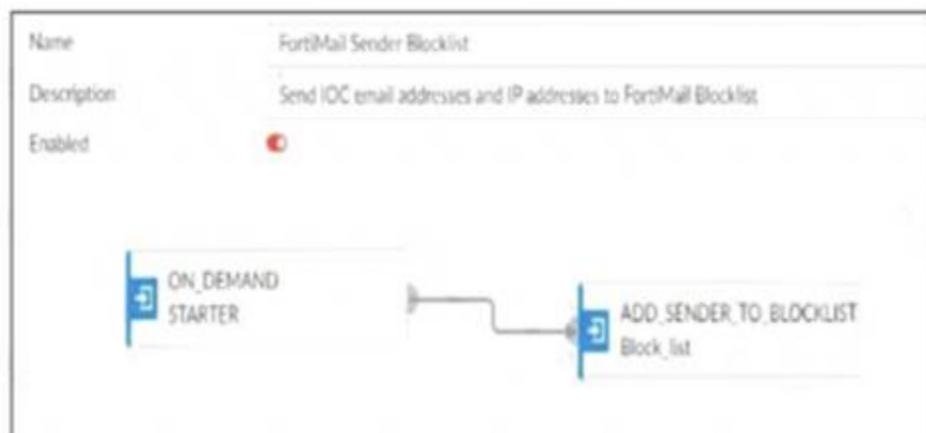
Fortinet Documentation on Playbook and Task Configuration.

Python error handling documentation for understandingValueError.

NEW QUESTION 5

Refer to the exhibits.

Playbook configuration



FortiMail connector actions

Configuration	Action	Description	Filters/Parameters
Enabled	ADD_SENDER_TO_BLOCKLIST	disard email received from the blocklis...	id: cmd:
Enabled	GET_EMAIL_STATISTICS	retrieve information of email message...	id: cmd:
Enabled	GET_SENDER_REPUTATION	retrieve information such as the sende...	id: ...

The FortiMail Sender Blocklist playbook is configured to take manual input and add those entries to the FortiMail abc. com domain-level block list. The playbook is configured to use a FortiMail connector and the ADD_SENDER_TO_BLOCKLIST action. Why is the FortiMail Sender Blocklist playbook execution failing?

- A. You must use the GET_EMAIL_STATISTICS action first to gather information about email messages.
- B. FortiMail is expecting a fully qualified domain name (FQDN).
- C. The client-side browser does not trust the FortiAnalyzer self-signed certificate.
- D. The connector credentials are incorrect

Answer: B

Explanation:

Understanding the Playbook Configuration:

The playbook "FortiMail Sender Blocklist" is designed to manually input email addresses or IP addresses and add them to the FortiMail block list. The playbook uses a FortiMail connector with the action ADD_SENDER_TO_BLOCKLIST.

Analyzing the Playbook Execution:

The configuration and actions provided show that the playbook is straightforward, starting with an ON_DEMAND STARTER and proceeding to the ADD_SENDER_TO_BLOCKLIST action.

The action description indicates it is intended to block senders based on email addresses or domains.

Evaluating the Options:

Option A: Using GET_EMAIL_STATISTICS is not required for the task of adding senders to a block list. This action retrieves email statistics and is unrelated to the block list configuration.

Option B: The primary reason for failure could be the requirement for a fully qualified domain name (FQDN). FortiMail typically expects precise information to ensure the correct entries are added to the block list.

Option C: The trust level of the client-side browser with FortiAnalyzer's self-signed certificate does not impact the execution of the playbook on FortiMail.

Option D: Incorrect connector credentials would result in an authentication error, but the problem described is more likely related to the format of the input data.

Conclusion:

The FortiMail Sender Blocklist playbook execution is failing because FortiMail is expecting a fully qualified domain name (FQDN).

References:

Fortinet Documentation on FortiMail Connector Actions.

Best Practices for Configuring FortiMail Block Lists.

NEW QUESTION 6

Refer to the exhibits.

Threat Hunting Monitor

Threat Action (2)	#	Application Service	Count	Sent (bytes)	Average Sent	Max Sent (bytes)
Threat Pattern (216)	1		251,400 (88%)			
Threat Name (54)	2	DNS	109,486 (30%)	9.1 MB	169.0 B	28.3 KB
Threat Type (8)	3	HTTP	4,521 (1%)	3.6 MB	1.2 KB	27.8 KB
File Hash (2)	4	HTTPS	1,026 (1%)	572.1 MB	578.3 KB	554.9 MB
File Name (2)	5	SSL	249 (1%)			
Application Process (2)	6	other	76 (1%)	10.2 KB	138.0 B	500.0 B
Application Name (32)	7	udp/443	58 (1%)	1019.8 KB	17.6 KB	17.6 KB
Application Service (21)	8	NTP	57 (1%)			

Threat Hunting Monitor

#	Date/Time	Event Message	Source IP	Destination IP
1	20:55:55		10.0.1.10	8.8.8.8
2	20:55:55	Connection Failed	10.0.1.10	8.8.8.8
3	20:55:55		10.0.1.10	8.8.8.8
4	20:55:55	Connection Failed	10.0.1.10	8.8.8.8
5	20:55:55		10.0.1.10	8.8.8.8
6	20:55:55	Connection Failed	10.0.1.10	8.8.8.8
7	20:55:55		10.0.1.10	8.8.8.8

What can you conclude from analyzing the data using the threat hunting module?

- A. Spearphishing is being used to elicit sensitive information.
- B. DNS tunneling is being used to extract confidential data from the local network.
- C. Reconnaissance is being used to gather victim identity information from the mail server.
- D. FTP is being used as command-and-control (C&C) technique to mine for data.

Answer: B

Explanation:

Understanding the Threat Hunting Data:

The Threat Hunting Monitor in the provided exhibits shows various application services, their usage counts, and data metrics such as sent bytes, average sent bytes, and maximum sent bytes.

The second part of the exhibit lists connection attempts from a specific source IP (10.0.1.10) to a destination IP (8.8.8.8), with repeated "Connection Failed" messages.

Analyzing the Application Services:

DNS is the top application service with a significantly high count (251,400) and notable sent bytes (9.1 MB).

This large volume of DNS traffic is unusual for regular DNS queries and can indicate the presence of DNS tunneling.

DNS Tunneling:

DNS tunneling is a technique used by attackers to bypass security controls by encoding data within DNS queries and responses. This allows them to extract data from the local network without detection.

The high volume of DNS traffic, combined with the detailed metrics, suggests that DNS tunneling might be in use.

Connection Failures to 8.8.8.8:

The repeated connection attempts from the source IP (10.0.1.10) to the destination IP (8.8.8.8) with connection failures can indicate an attempt to communicate with an external server.

Google DNS (8.8.8.8) is often used for DNS tunneling due to its reliability and global reach.

Conclusion:

Given the significant DNS traffic and the nature of the connection attempts, it is reasonable to conclude that DNS tunneling is being used to extract confidential data from the local network.

Why Other Options are Less Likely:

Spearphishing (A): There is no evidence from the provided data that points to spearphishing attempts, such as email logs or phishing indicators.

Reconnaissance (C): The data does not indicate typical reconnaissance activities, such as scanning or probing mail servers.

FTP C&C (D): There is no evidence of FTP traffic or command-and-control communications using FTP in the provided data.

References:

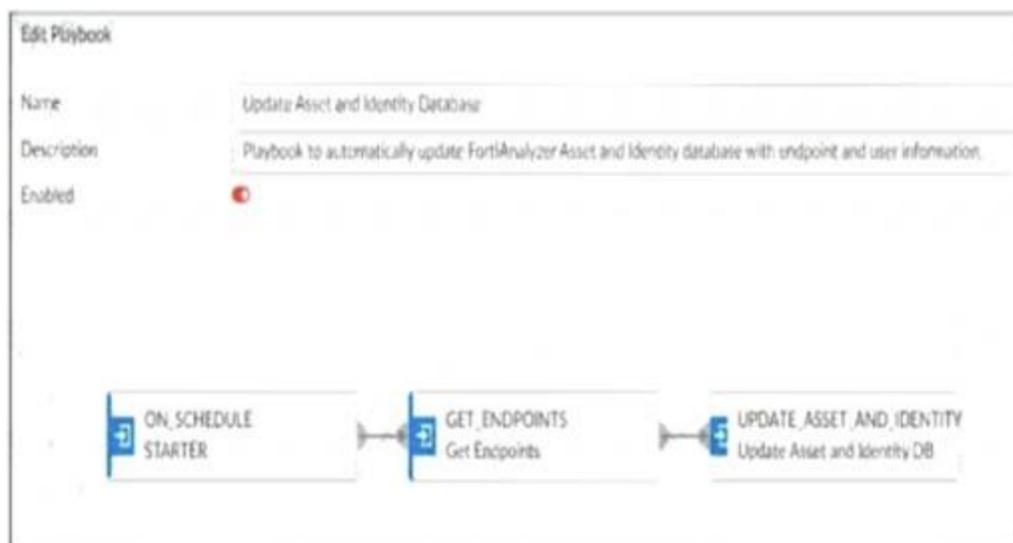
SANS Institute: "DNS Tunneling: How to Detect Data Exfiltration and Tunneling Through DNS Queries" SANS DNS Tunneling

OWASP: "DNS Tunneling" OWASP DNS Tunneling

By analyzing the provided threat hunting data, it is evident that DNS tunneling is being used to exfiltrate data, indicating a sophisticated method of extracting confidential information from the network.

NEW QUESTION 7

Refer to the exhibit.



Which two options describe how the Update Asset and Identity Database playbook is configured? (Choose two.)

- A. The playbook is using a local connector.
- B. The playbook is using a FortiMail connector.
- C. The playbook is using an on-demand trigger.
- D. The playbook is using a FortiClient EMS connector.

Answer: AD

Explanation:

Understanding the Playbook Configuration:

The playbook named "Update Asset and Identity Database" is designed to update the FortiAnalyzer Asset and Identity database with endpoint and user information.

The exhibit shows the playbook with three main components: ON_SCHEDULE STARTER, GET_ENDPOINTS, and UPDATE_ASSET_AND_IDENTITY.

Analyzing the Components:

ON_SCHEDULE STARTER: This component indicates that the playbook is triggered on a schedule, not on-demand.

GET_ENDPOINTS: This action retrieves information about endpoints, suggesting it interacts with an endpoint management system.

UPDATE_ASSET_AND_IDENTITY: This action updates the FortiAnalyzer Asset and Identity database with the retrieved information.

Evaluating the Options:

Option A: The actions shown in the playbook are standard local actions that can be executed by the FortiAnalyzer, indicating the use of a local connector.

Option B: There is no indication that the playbook uses a FortiMail connector, as the tasks involve endpoint and identity management, not email.

Option C: The playbook is using an "ON_SCHEDULE" trigger, which contradicts the description of an on-demand trigger.

Option D: The action "GET_ENDPOINTS" suggests integration with an endpoint management system, likely FortiClient EMS, which manages endpoints and retrieves information from them.

Conclusion:

The playbook is configured to use a local connector for its actions.

It interacts with FortiClient EMS to get endpoint information and update the FortiAnalyzer Asset and Identity database.

References:

Fortinet Documentation on Playbook Actions and Connectors.

FortiAnalyzer and FortiClient EMS Integration Guides.

NEW QUESTION 8

Which two statements about the FortiAnalyzer Fabric topology are true? (Choose two.)

- A. Downstream collectors can forward logs to Fabric members.
- B. Logging devices must be registered to the supervisor.
- C. The supervisor uses an API to store logs, incidents, and events locally.
- D. Fabric members must be in analyzer mode.

Answer: BD

Explanation:

The FortiAnalyzer Fabric topology is designed to centralize logging and analysis across multiple devices in a network.

It involves a hierarchy where the supervisor node manages and coordinates with other Fabric members.

Analyzing the Options:

Option A: Downstream collectors forwarding logs to Fabric members is not a typical configuration. Instead, logs are usually centralized to the supervisor.

Option B: For effective management and log centralization, logging devices must be registered to the supervisor. This ensures proper log collection and coordination.

Option C: The supervisor does not primarily use an API to store logs, incidents, and events locally. Logs are stored directly in the FortiAnalyzer database.

Option D: For the Fabric topology to function correctly, all Fabric members need to be in analyzer mode. This mode allows them to collect, analyze, and forward logs appropriately within the topology.

Conclusion:

The correct statements regarding the FortiAnalyzer Fabric topology are that logging devices must be registered to the supervisor and that Fabric members must be in analyzer mode.

References:

Fortinet Documentation on FortiAnalyzer Fabric Topology.

Best Practices for Configuring FortiAnalyzer in a Fabric Environment.

NEW QUESTION 9

Refer to the exhibit.

Events

Event ID	Event Status	Event Type	Count	Severity	First Occurrence	Last Update	Handler
Device offline (1)	Event	Event	1	Medium	4 minutes ago	4 minutes ago	Local Device Event
FortiMail (400)	Unhandled	Email Filter	400	High	2 minutes ago	a minute ago	SOC SMTP Enumeration Data Handler
devname:FortiMail from:en	Unhandled	Email Filter	1	High	2024-03-13 18:56:52	2024-03-13 18:57:03	SOC SMTP Enumeration Data Handler
devname:FortiMail from:en	Unhandled	Email Filter	1	High	2024-03-13 18:56:52	2024-03-13 18:57:03	SOC SMTP Enumeration Data Handler
devname:FortiMail from:en	Unhandled	Email Filter	1	High	2024-03-13 18:56:52	2024-03-13 18:57:03	SOC SMTP Enumeration Data Handler
devname:FortiMail from:en	Unhandled	Email Filter	1	High	2024-03-13 18:56:52	2024-03-13 18:57:03	SOC SMTP Enumeration Data Handler
devname:FortiMail from:en	Unhandled	Email Filter	1	High	2024-03-13 18:56:52	2024-03-13 18:57:03	SOC SMTP Enumeration Data Handler
devname:FortiMail from:en	Unhandled	Email Filter	1	High	2024-03-13 18:56:52	2024-03-13 18:57:03	SOC SMTP Enumeration Data Handler
devname:FortiMail from:en	Unhandled	Email Filter	1	High	2024-03-13 18:56:52	2024-03-13 18:57:03	SOC SMTP Enumeration Data Handler
devname:FortiMail from:en	Unhandled	Email Filter	1	High	2024-03-13 18:56:52	2024-03-13 18:57:03	SOC SMTP Enumeration Data Handler
devname:FortiMail from:en	Unhandled	Email Filter	1	High	2024-03-13 18:56:52	2024-03-13 18:57:03	SOC SMTP Enumeration Data Handler
devname:FortiMail from:en	Unhandled	Email Filter	1	High	2024-03-13 18:56:51	2024-03-13 18:57:03	SOC SMTP Enumeration Data Handler
devname:FortiMail from:en	Unhandled	Email Filter	1	High	2024-03-13 18:56:51	2024-03-13 18:57:03	SOC SMTP Enumeration Data Handler
devname:FortiMail from:en	Unhandled	Email Filter	1	High	2024-03-13 18:56:51	2024-03-13 18:57:03	SOC SMTP Enumeration Data Handler
devname:FortiMail from:en	Unhandled	Email Filter	1	High	2024-03-13 18:56:51	2024-03-13 18:57:03	SOC SMTP Enumeration Data Handler

Event Handler

Status: ●

Name: SOC SMTP Enumeration Data Handler

Description:

You notice that the custom event handler you configured to detect SMTP reconnaissance activities is creating a large number of events. This is overwhelming your notification system. How can you fix this?

- A. Increase the trigger count so that it identifies and reduces the count triggered by a particular group.
- B. Disable the custom event handler because it is not working as expected.
- C. Decrease the time range that the custom event handler covers during the attack.
- D. Increase the log field value so that it looks for more unique field values when it creates the event.

Answer: A

Explanation:

Understanding the Issue:

The custom event handler for detecting SMTP reconnaissance activities is generating a large number of events.

This high volume of events is overwhelming the notification system, leading to potential alert fatigue and inefficiency in incident response.

Event Handler Configuration:

Event handlers are configured to trigger alerts based on specific criteria.

The frequency and volume of these alerts can be controlled by adjusting the trigger conditions.

Possible Solutions:

* A. Increase the trigger count so that it identifies and reduces the count triggered by a particular group:

By increasing the trigger count, you ensure that the event handler only generates alerts after a higher threshold of activity is detected.

This reduces the number of events generated and helps prevent overwhelming the notification system.

Selected as it effectively manages the volume of generated events.

* B. Disable the custom event handler because it is not working as expected:

Disabling the event handler is not a practical solution as it would completely stop monitoring for SMTP reconnaissance activities.

Not selected as it does not address the issue of fine-tuning the event generation.

* C. Decrease the time range that the custom event handler covers during the attack:

Reducing the time range might help in some cases, but it could also lead to missing important activities if the attack spans a longer period.

Not selected as it could lead to underreporting of significant events.

* D. Increase the log field value so that it looks for more unique field values when it creates the event:

Adjusting the log field value might refine the event criteria, but it does not directly control the volume of alerts.

Not selected as it is not the most effective way to manage event volume.

Implementation Steps:

Step 1: Access the event handler configuration in FortiAnalyzer.

Step 2: Locate the trigger count setting within the custom event handler for SMTP reconnaissance.

Step 3: Increase the trigger count to a higher value that balances alert sensitivity and volume.

Step 4: Save the configuration and monitor the event generation to ensure it aligns with expected levels.

Conclusion:

By increasing the trigger count, you can effectively reduce the number of events generated by the custom event handler, preventing the notification system from being overwhelmed.

References:

Fortinet Documentation on Event Handlers and Configuration FortiAnalyzer Administration Guide

Best Practices for Event Management Fortinet Knowledge Base

By increasing the trigger count in the custom event handler, you can manage the volume of generated events and prevent the notification system from being overwhelmed.

NEW QUESTION 10

.....

Thank You for Trying Our Product

* 100% Pass or Money Back

All our products come with a 90-day Money Back Guarantee.

* One year free update

You can enjoy free update one year. 24x7 online support.

* Trusted by Millions

We currently serve more than 30,000,000 customers.

* Shop Securely

All transactions are protected by VeriSign!

100% Pass Your FCSS_SOC_AN-7.4 Exam with Our Prep Materials Via below:

https://www.certleader.com/FCSS_SOC_AN-7.4-dumps.html