PassQueen
a.k.a dumpsqueen.com

Riverbed Certified Solutions Associate - Application Performance Management

Riverbed 401-01

Version Demo

Total Demo Questions: 8

Total Premium Questions: 80

Buy Premium PDF

https://passqueen.com

support@passqueen.com

Topic Break Down

Topic	No. of Questions
Topic 1, Application Performance Management Fundamentals	37
Topic 2, SteelCentral AppResponse	27
Topic 3, SteelCentral NetProfiler	4
Topic 4, SteelCentral Portal	12
Total	80

QUESTION NO: 1

Which statements correctly compare packet data and flow data for application-performance monitoring? (Choose two)

A. Packet data supports detailed response-time and protocol analysis

B. Flow data is well suited to scalable network-wide traffic visibility

C. Flow data preserves every TCP packet and payload

D. Packet data cannot identify retransmissions

ANSWER: A B

Explanation:

Packet data supports detailed response-time and protocol analysis and **flow data is well suited to scalable network-wide traffic visibility** are correct. Packet capture preserves the sequence, timing, and protocol details needed to investigate conditions such as retransmissions, resets, and request-response behavior. This makes it valuable for focused troubleshooting.

Flow telemetry summarizes communications into records, making it easier to collect and analyze activity across a large number of network devices and sites. It provides broad visibility into conversations and traffic volumes, but does not replace packet data for deep transaction-level diagnosis. See Riverbed NetProfiler, Riverbed AppResponse, and the Riverbed Documentation portal.

QUESTION NO: 2

In an AppResponse Response Time Composition Chart, a large server-time component most directly indicates that the transaction spent significant time:

A. Waiting for the server to respond

B. Resolving a client DNS query

C. Writing packet data to the AppResponse archive

D. Exporting NetFlow records

ANSWER: A

Explanation:

Waiting for the server to respond is correct. The server-time portion of response-time composition represents the delay after the request is delivered to the server and before the response begins returning to the client. A large value focuses investigation on server-side application processing, downstream dependencies, or other conditions delaying the response.

This does not prove that the root cause is the server operating system itself; the delay can be caused by application logic, database activity, a remote service call, or resource contention. However, response-time composition helps separate this server-associated delay from network transport effects. Riverbed AppResponse information is available at the Riverbed AppResponse product page.

QUESTION NO: 3

For what types of applications does AppResponse support response-time monitoring?

A. Applications with predefined decode support

B. Applications that are preloaded for analysis into AppResponse

C. Applications that are instrumented with agents

D. Any applications running on TCP

ANSWER: D

Explanation:

AppResponse supports response-time monitoring for any application that runs over TCP. This capability is based on passive observation of the TCP conversation rather than requiring application code changes or an installed agent. By examining the bidirectional packet flow, AppResponse can identify request and response data exchanges and calculate response-time behavior for the TCP application traffic it sees. This makes the capability useful not only for well-known, decoded application protocols, but also for internally developed, custom, or otherwise unclassified TCP applications, provided that the appliance observes both directions of the relevant traffic. Riverbed describes AppResponse as a network-performance monitoring platform that analyzes application traffic and provides response-time visibility from packet data. Its documentation resources also cover the platform's TCP- and packet-based analysis capabilities. See the Riverbed AppResponse product page and the Riverbed documentation portal for product and configuration documentation.

QUESTION NO: 4

The privilege level of a user that can make configuration changes in AppResponse except for changes in the User Admin Manager, the web interface or the serial console is:

A. Administrative

B. Basic

C. Standard

D. Restricted

ANSWER: C

Explanation:

Standard is the AppResponse privilege level intended for users who need to perform operational configuration work without receiving full appliance-administration authority. A Standard user can make the applicable AppResponse configuration changes, allowing them to support day-to-day monitoring and analysis workflows. The role deliberately excludes access to the User Admin Manager, web-interface administration, and serial-console administration. Those areas control user access and core appliance-management settings, so restricting them preserves separation between operational responsibilities and security-sensitive administrative responsibilities.

This distinction is important when assigning least-privilege access: personnel responsible for configuring monitoring functions can receive the capabilities needed for their work, while account management and interface or console administration remain under a more privileged role. Riverbed's documentation resources organize product guidance, including AppResponse administration and user-access material, in the Riverbed Documentation Portal. Product documentation and support resources are also available through Riverbed Support.

QUESTION NO: 5

In AppResponse, alerts can be displayed on the Alert Dashboard in these sections. (Choose two)

A. Client Issues

B. Browser Issues

C. Webpage Issues

D. Server Issues

ANSWER: A D

Explanation:

AppResponse organizes Alert Dashboard content into problem-domain sections so that operators can quickly identify which side of an application transaction is associated with an active condition. **Client Issues** is a valid Alert Dashboard section for alerts associated with client-side behavior and the user-facing end of observed application activity. This grouping helps an operator focus investigation on conditions affecting clients, rather than starting from an undifferentiated list of alerts.

Server Issues is also a valid Alert Dashboard section. It presents alerts associated with server-side conditions, allowing operations staff to identify and investigate application infrastructure or server-related symptoms from the same dashboard. These two sections reflect AppResponse's high-level alert triage workflow: separate the client and server perspectives first, then drill into the relevant alert details and supporting performance data. Riverbed product documentation is version-specific, so the AppResponse documentation matching the deployed release should be used when verifying dashboard labels and available alert views. See the Riverbed Support portal for product documentation and software resources, and Riverbed's documentation page for documentation access.

QUESTION NO: 6

Why is a common time source recommended for all hosts participating in an AppInternals deployment?

A. It supports accurate correlation of transaction segments across tiers

B. It increases packet-storage capacity

C. It removes the need to configure thresholds

D. It automatically creates Portal dashboards

ANSWER: A

Explanation:

It supports accurate correlation of transaction segments across tiers is correct. AppInternals uses timestamps from agents monitoring different application tiers when it detects and stitches related activity into an end-to-end transaction. Clock differences can cause related events to appear out of sequence or outside expected correlation timing.

Synchronizing the Analysis Server and monitored hosts through a common Network Time Protocol source helps preserve accurate transaction timing and cross-tier correlation. See the Network Time Protocol Version 4 specification and the Riverbed Support portal.

QUESTION NO: 7

Which of the following are typical questions answered by AppInternals? (Select three)

A. "Which classes and methods are slowing down my application?"

B. "Which packets are related to my application transaction?"

C. "What servers are used by a given transaction instance?"

D. Which routers are being traversed by my application?"

E. "Which transactions were run by a given user?"

ANSWER: A B E

Explanation:

AppInternals is designed for transaction-level application performance troubleshooting, combining application-aware visibility with detailed diagnostic evidence. "Which classes and methods are slowing down my application?" is a core use case because AppInternals traces application execution and exposes time spent in code-level components, helping an administrator isolate costly methods within a transaction. "Which packets are related to my application transaction?" is also a

typical diagnostic question: transaction analysis correlates application activity with the relevant network communication so that the transaction can be investigated in context rather than as an unrelated packet stream. “Which transactions were run by a given user?” is supported through transaction and user/session-oriented visibility, allowing investigations to begin with the affected user experience and then drill into individual transaction instances and their execution details. Together, these questions reflect AppInternals’ purpose: connecting user activity, transactions, application code behavior, and associated network evidence to shorten root-cause analysis. Riverbed describes its application-performance portfolio and related observability capabilities at Riverbed Products; product documentation and release resources are available through the Riverbed Support portal.

QUESTION NO: 8

NetProfiler flow records can provide which information for network-wide traffic analysis? (Choose two)

A. Source and destination communication details

B. Traffic volume and rate information

C. Complete application payload contents

D. Code-level Java method execution time

ANSWER: A B

Explanation:

Source and destination communication details are available from flow records, which identify the endpoints participating in a network conversation. This helps analysts determine which systems are communicating across the monitored environment.

Traffic volume and rate information is also available because flow records include counters and timing information used to calculate bytes, packets, and throughput. NetProfiler aggregates this data from flow-exporting infrastructure to provide scalable visibility across a large network. Unlike full packet capture, flow records do not preserve the complete payload or every packet exchange. Riverbed NetProfiler information is available at the Riverbed NetProfiler product page.