DUMPSARENA

Splunk Enterprise Certified Architect

Splunk SPLK-2002

Version Demo

Total Demo Questions: 22

Total Premium Questions: 228

Buy Premium PDF

https://dumpsarena.co

sales@dumpsarena.co

sales@dumpsarena.co
dumpsarena.co

Topic Break Down

Topic	No. of Questions
Topic 1, Splunk Deployment Planning	44
Topic 2, Splunk Deployment Management	35
Topic 3, Indexer Clustering	53
Topic 4, Search Head Clustering	48
Topic 5, Data Collection	23
Topic 6, Forwarder Management	22
Topic 7, Mix Questions	3
Total	228

QUESTION NO: 1

Which of the following clarification steps should be taken if apps are not appearing on a deployment client? (Select all that apply.)

A. Check serverclass.conf of the deployment server.

B. Check deploymentclient.conf of the deployment client.

C. Check the content of SPLUNK_HOME/etc/deployment-apps of the deployment server.

D. Search for relevant events in splunkd.log of the deployment server.

ANSWER: A B C D

Explanation:

Checking `serverclass.conf` on the deployment server is essential because server classes define both the client membership criteria and the deployment apps assigned to matching clients. Confirming the deployment client's `deploymentclient.conf` is equally important: it establishes the deployment server target and the client identity information used when evaluating server-class filters. The deployment server must store deployable app content under `$SPLUNK_HOME/etc/deployment-apps`; this is the source directory from which the deployment server distributes app payloads to clients. Finally, reviewing the deployment server's `splunkd.log` provides operational evidence of phone-home activity, server-class matching, bundle generation, and deployment processing. Together, these checks validate the configuration, app location, client-to-server connectivity, and processing path required for an app to be delivered. Splunk documents deployment-app storage and server-class assignment in its Deploy apps to deployment clients guidance, and documents deployment-client configuration in Configure deployment clients.

QUESTION NO: 2

What types of files exist in a bucket within a clustered index? (select all that apply)

A. Inside a replicated bucket, there is only rawdata.

B. Inside a searchable bucket, there is only tsidx.

C. Inside a searchable bucket, there is tsidx and rawdata.

D. Inside a replicated bucket, there is both tsidx and rawdata.

ANSWER: C D

Explanation:

A searchable bucket contains both the compressed source-event data in `rawdata` and the time-series index files, commonly called `tsidx` files. The raw data provides the original event content, while the tsidx files provide the indexed structures that enable Splunk Enterprise to efficiently locate events during a search. Therefore, the statement that a searchable bucket contains tsidx and rawdata accurately describes the required contents of a searchable bucket copy.

A replicated bucket can also contain both tsidx and rawdata when that replicated copy is maintained as a searchable copy. In an indexer cluster, bucket copies are distributed among peer nodes to satisfy the replication factor, and the search factor determines how many copies are searchable. A searchable replicated copy must include both the event data and its index files so that search heads can use that peer to execute searches. This allows the cluster to retain availability for search as well as data redundancy when peer nodes become unavailable.

Splunk's indexer-cluster documentation describes searchable bucket copies as containing both raw data and index files, and explains how replication and search factors govern bucket-copy placement and searchability. See How indexer clustering works and Replication factor and search factor.

QUESTION NO: 3

Which setting in `inputs.conf` prevents a monitored file input from indexing files older than a specified number of seconds?

A. ignoreOlderThan

B. maxFileAge

C. skipOldFiles

D. monitorAgeLimit

ANSWER: A

Explanation:

`ignoreOlderThan` prevents a monitor input from reading files whose modification time is older than the configured number of seconds. This is useful when a directory contains historical files that should not be ingested when an input is first enabled.

The setting applies to file and directory monitoring inputs in `inputs.conf`. It should be selected carefully because files older than the configured threshold are skipped, even if their content would otherwise match the monitor stanza. See the inputs.conf specification.

QUESTION NO: 4

Determining data capacity for an index is a non-trivial exercise. Which of the following are possible considerations that would affect daily indexing volume? (select all that apply)

A. Average size of event data.

B. Number of data sources.

C. Peak data rates.

D. Number of concurrent searches on data.

ANSWER: A B C

Explanation:

“Average size of event data,” “Number of data sources,” and “Peak data rates” are appropriate considerations when estimating indexing capacity. Daily indexing volume is fundamentally based on the amount of incoming raw data, so the average number of bytes in each event directly affects the number of bytes Splunk must parse, index, and store. Event volume and event size together determine the raw-data contribution of a given input.

“Number of data sources” is also relevant because each onboarded source can contribute a distinct stream of events. A complete estimate accounts for the event rate and average event size from every source, then aggregates those values to establish the expected daily total. Sources can also have substantially different data patterns, so grouping all data under a single assumed event profile can produce an inaccurate estimate.

“Peak data rates” are essential to capacity planning because ingestion is rarely uniform throughout the day. Splunk infrastructure must be able to accept, parse, and index bursts at the expected peak rate while still accommodating the projected daily data total. Assessing both sustained volume and burst behavior helps ensure that indexer resources and ingestion paths are sized for real operating conditions. Splunk’s capacity-planning guidance emphasizes evaluating data volume, ingestion rates, and workload characteristics as part of deployment sizing. See the Splunk Enterprise Capacity Planning Manual and Splunk guidance for planning data storage.

QUESTION NO: 5

A Splunk environment collecting 10 TB of data per day has 50 indexers and 5 search heads. A single-site indexer cluster will be implemented. Which of the following is a best practice for added data resiliency?

A. Set the Replication Factor to 49.

B. Set the Replication Factor based on allowed indexer failure.

C. Always use the default Replication Factor of 3.

D. Set the Replication Factor based on allowed search head failure.

ANSWER: B

Explanation:

Set the Replication Factor based on allowed indexer failure. In a single-site indexer cluster, the replication factor determines how many total copies of each bucket Splunk maintains on different peer nodes. This setting is central to data resiliency because it establishes how many peer-node failures the cluster can tolerate while retaining bucket copies. The appropriate value is driven by the organization's availability requirements and its planned failure tolerance, rather than by a fixed default or by the number of search heads. For example, a replication factor of 3 creates three bucket copies and can tolerate the loss of up to two peer nodes without losing all copies of a bucket. The cluster must also have enough peer nodes to satisfy the configured replication factor, which is easily accommodated by a 50-indexer deployment. Splunk recommends configuring replication and search factors according to the desired level of data availability and search availability. While the replication factor protects the presence of bucket copies, the related search factor specifies how many copies must be searchable; together, these settings should be sized to meet the resiliency objectives of the indexer tier. See Splunk's replication factor documentation and its overview of indexer clusters.

QUESTION NO: 6

(Which of the following must be included in a deployment plan?)

A. Future topology diagrams of the IT environment.

B. A comprehensive list of stakeholders, either direct or indirect.

C. Current logging details and data source inventory.

D. Business continuity and disaster recovery plans.

ANSWER: C

Explanation:

Current logging details and data source inventory are essential components of a Splunk deployment plan because they establish the technical scope on which the deployment is designed. An architect needs to identify each intended data source, its collection method, estimated daily volume, event format, expected growth, retention needs, and any availability or compliance requirements. These inputs drive capacity estimates for indexing throughput and storage, determine appropriate forwarder and input configurations, and support decisions about index design and licensing. They also help ensure that the proposed architecture can reliably ingest the required data and make it searchable for the intended use cases. Splunk's capacity-planning guidance emphasizes understanding anticipated data volume, data characteristics, and retention as prerequisites for sizing a deployment. Its data-ingestion documentation likewise establishes that knowing the source and type of data is fundamental to configuring inputs and event processing appropriately. A complete, current inventory therefore gives the deployment plan a measurable basis for infrastructure design, implementation sequencing, and operational readiness. See Splunk's Introduction to capacity planning and What data can Splunk Enterprise index?.

QUESTION NO: 7

Which of the following statements about integrating with third-party systems is true? (Select all that apply.)

A. A Hadoop application can search data in Splunk.

B. Splunk can search data in the Hadoop File System (HDFS).

C. You can use Splunk alerts to provision actions on a third-party system.

D. You can forward data from Splunk forwarder to a third-party system without indexing it first.

ANSWER: A B C D

Explanation:

All listed integration capabilities are valid in the appropriate Splunk deployment and product context. A Hadoop application can search data in Splunk because Splunk's Hadoop integration provides mechanisms for Hadoop workloads to access Splunk data and search results. Splunk can also search data in the Hadoop File System (HDFS) through Splunk's Hadoop-related integration capabilities, enabling analysis of Hadoop-resident data without requiring it to be treated solely as conventional locally indexed Splunk data. Splunk documentation for Hadoop Connect describes bidirectional integration patterns between Splunk and Hadoop, including using Splunk data with Hadoop and bringing Hadoop data into Splunk workflows: Splunk Hadoop Connect documentation.

You can use Splunk alerts to provision actions on a third-party system because alert actions can invoke external workflows, including scripts, webhooks, and supported integrations. This allows a saved search or correlation condition to initiate an operational response outside Splunk. Splunk can also forward event data to external destinations without first indexing that data locally when a forwarding architecture and supported output mechanism are configured for that purpose. This is consistent with Splunk's routing and forwarding model, in which a forwarder can direct data onward rather than retaining it in a local index. See Configure alert actions.

QUESTION NO: 8

What log file would you search to verify if you suspect there is a problem interpreting a regular expression in a monitor stanza?

A. btool.log

B. metrics.log

C. splunkd.log

D. tailing_processor.log

ANSWER: C

Explanation:

`splunkd.log` is the appropriate log to search because it is Splunk Enterprise's principal operational log for the `splunkd` service and its processing components. Monitor stanzas in `inputs.conf` are handled by Splunk's file-monitoring and tailing processes; when Splunk initializes, validates, or runs those inputs, messages from relevant components, including TailingProcessor-related activity, are recorded in `$SPLUNK_HOME/var/log/splunk/splunkd.log`. A regular expression that cannot be compiled or interpreted as expected can therefore produce diagnostic messages there, often containing the affected stanza, pattern, source path, component name, or an error describing the expression issue. Searching this log around the time the input was deployed or Splunk was restarted is especially useful. In Splunk, `inputs.conf` defines file and directory monitoring behavior, including monitor stanzas and their settings. The Splunk Enterprise troubleshooting documentation also identifies `splunkd.log` as a core source for diagnosing service and data-input processing behavior. See the inputs.conf specification and Splunk's guidance on useful log files for troubleshooting.

QUESTION NO: 9

Which of the following actions should be performed through a search head cluster deployer? (Select all that apply.)

A. Deploy an app containing search-time knowledge objects to all members.

B. Deploy app-level props.conf and transforms.conf files to all members.

C. Manually copy app files to each cluster member.

D. Replicate runtime knowledge object changes made by users.

ANSWER: A B

Explanation:

The deployer is used to distribute apps and deployer-managed configuration bundles consistently to all members of a search head cluster. An app containing search-time knowledge objects, such as dashboards, macros, event types, tags, field extractions, and lookup files, should be installed on the deployer and pushed to the cluster through the deployer workflow. This ensures that all members receive the same app content.

Configuration files packaged within an app, including search-time `props.conf` and `transforms.conf` content, should also be deployed through the deployer when the configuration must be common to all cluster members. Administrators should not manually copy files to individual members because that creates configuration drift. Runtime changes to replicated knowledge objects are synchronized by the search head cluster itself and are not distributed by the deployer. See Splunk's use the deployer to distribute apps and configuration updates documentation.

QUESTION NO: 10

In splunkd. log events written to the _internal index, which field identifies the specific log channel?

A. component

B. source

C. sourcetype

D. channel

ANSWER: A

Explanation:

component is the field that identifies the specific splunkd logging channel or subsystem responsible for an event. Events from `splunkd.log` are indexed in `_internal` and Splunk extracts fields that make it possible to isolate operational activity by the part of splunkd that produced the message. For example, searches can use `component=SearchParser`, `component=Metrics`, or another component value to focus troubleshooting on a particular service or functional area. This is especially useful because `splunkd.log` contains messages from many internal processes and logging categories in a single log file. The component value is emitted as part of the splunkd log message format and provides the meaningful channel-level identification used when investigating Splunk Enterprise behavior, errors, performance messages, and startup activity. Splunk's documentation describes splunkd logging as a primary source for diagnosing Splunk Enterprise operation and documents the internal logs available for troubleshooting. See What Splunk logs about itself and Use splunkd.log.

QUESTION NO: 11

Which of the following should be done when installing Enterprise Security on a Search Head Cluster? (Select all that apply.)

A. Install Enterprise Security on the deployer.

B. Install Enterprise Security on a staging instance.

C. Copy the Enterprise Security configurations to the deployer.

D. Use the deployer to deploy Enterprise Security to the cluster members.

ANSWER: A D

Explanation:

Install Enterprise Security on the deployer. The deployer is the administrative Splunk Enterprise instance used to manage and distribute apps and app updates across the members of a search head cluster. Installing the Enterprise Security app there creates the deployment source for a consistent app bundle, rather than treating individual cluster members as separate installation targets.

Use the deployer to deploy Enterprise Security to the cluster members. Deploying from the deployer distributes the Enterprise Security app and its required contents to every search head cluster member, helping ensure that the cluster runs the same app version and configuration. This is the standard search head cluster app-distribution workflow and is especially important for an app such as Enterprise Security, whose knowledge objects, dashboards, correlation-search content, and supporting configurations must be consistently available throughout the cluster. Splunk documents the Enterprise Security search head cluster installation process as installing the app on the deployer and then using the deployer to push it to the cluster. See Install Enterprise Security in a search head cluster and Use the deployer to distribute apps and configuration updates.

QUESTION NO: 12

Which of the following statements describe a Search Head Cluster (SHC) captain? (Select all that apply.)

A. Is the job scheduler for the entire SHC.

B. Manages alert action suppressions (throttling).

C. Synchronizes the member list with the KV store primary.

D. Replicates the SHC's knowledge bundle to the search peers.

ANSWER: A B C D

Explanation:

The SHC captain is the cluster's coordinating member and has several centralized responsibilities. "Is the job scheduler for the entire SHC" is correct because the captain schedules cluster-wide scheduled searches and assigns them to eligible members, helping ensure that scheduled workloads are not run redundantly by multiple search heads. "Manages alert action suppressions (throttling)" is also correct: the captain coordinates alert-suppression state so that a throttled alert action is consistently suppressed across the cluster rather than firing independently from different members.

"Synchronizes the member list with the KV store primary" describes another captain responsibility. The captain maintains cluster membership information and communicates relevant membership changes to the KV store primary, supporting the KV store's operation within the search head cluster. Finally, "Replicates the SHC's knowledge bundle to the search peers" is correct because the captain coordinates distribution of the knowledge bundle needed by search peers to execute distributed searches using the search head cluster's applicable knowledge objects and configurations. These captain duties allow SHC members to provide distributed, highly available search-head services while maintaining consistent scheduling, alerting, membership, and search-peer behavior. See Splunk's Search head cluster architecture documentation and How search head clustering works.

QUESTION NO: 13

Which Splunk server role regulates the functioning of indexer cluster?

A. Indexer

B. Deployer

C. Master Node

D. Monitoring Console

ANSWER: C

Explanation:

The cluster manager, formerly called the master node, regulates an indexer cluster. It is the central control plane for the cluster and maintains the configuration and state information needed for peer indexers to operate as a coordinated group. Its responsibilities include registering peer nodes, monitoring their status, coordinating replication and searchable-copy activities, and directing data rebalancing when necessary. It also manages the cluster's replication factor and search factor

policies, helping ensure that data copies are maintained and that sufficient searchable copies are available across the peer nodes. This role does not ordinarily perform the indexing workload of a peer indexer; instead, it coordinates the peer nodes that store and service clustered data. Splunk renamed the master node role to cluster manager in newer documentation, so “Master Node” remains the appropriate answer for terminology used by this question, while “cluster manager” is the current preferred name. See Splunk’s indexer clustering overview and its cluster operations documentation.

QUESTION NO: 14

To activate replication for an index in an indexer cluster, what attribute must be configured in indexes.conf on all peer nodes?

A. repFactor = 0

B. replicate = 0

C. repFactor = auto

D. replicate = auto

ANSWER: C

Explanation:

`repFactor = auto` is the required index-level setting to enable replication for a specific index in a Splunk indexer cluster. This setting belongs in the index’s stanza in `indexes.conf` and must be present consistently on every peer node. The value `auto` tells Splunk to apply the cluster-wide replication factor configured by the cluster manager, rather than setting an independent replication count for that individual index. This lets the index participate in the cluster’s normal bucket replication process, ensuring that copies of its data are distributed across peers according to the cluster configuration.

Splunk’s indexer-cluster documentation specifies that peer nodes require matching index configurations and identifies `repFactor = auto` as the setting that makes an index replicated. The cluster manager’s replication factor then governs how many copies of each bucket are maintained, while the index configuration determines that replication is enabled for that index. See Splunk’s Configure peer indexes documentation and the indexes.conf specification for the index configuration details.

QUESTION NO: 15

(What are the possible values for the mode attribute in server.conf for a Splunk server in the [clustering] stanza?)

A. [clustering] mode = peer

B. [clustering] mode = searchhead

C. [clustering] mode = deployer

D. [clustering] mode = manager

ANSWER: A B D

Explanation:

The valid `mode` values in the `[clustering]` stanza are `peer`, `searchhead`, and `manager`. The `[clustering] mode = peer` setting assigns an indexer-cluster peer role to the instance. A peer receives indexed data, maintains replicated data copies according to the cluster’s replication and search factors, and participates in distributed search.

The `[clustering] mode = searchhead` setting configures a search head to participate in searches across an indexer cluster. This role enables the search head to communicate with the cluster manager and dispatch searches to the peer nodes that hold the relevant data.

The `[clustering] mode = manager` setting configures the instance as the cluster manager. The manager coordinates cluster membership and peer activity, maintains bucket-placement and replication state, and manages the cluster's data-availability behavior. In older Splunk terminology, this role was called the cluster master; current documentation uses "cluster manager." These roles are configured in `server.conf` as part of the indexer-clustering deployment configuration. See Splunk's Configure an indexer cluster documentation and the server.conf specification.

QUESTION NO: 16

A technical add-on contains index-time `props.conf` and `transforms.conf` settings that must be deployed consistently to every peer in an indexer cluster. Where should the app be placed on the cluster manager before distributing the cluster bundle?

A. `$SPLUNK_HOME/etc/apps`

B. `$SPLUNK_HOME/etc/deployment-apps`

C. `$SPLUNK_HOME/etc/manager-apps`

D. `$SPLUNK_HOME/etc/shcluster/apps`

ANSWER: C

Explanation:

The app should be placed in `$SPLUNK_HOME/etc/manager-apps` on the cluster manager. The cluster manager uses this directory as the source for configuration bundles that it distributes to peer nodes. This provides a consistent method for deploying index-time parsing and routing configuration across the cluster.

After placing the app in the manager-apps directory, the administrator applies the cluster bundle from the cluster manager. The resulting configuration is distributed to the peer nodes, where parsing and indexing occur. Deploying the same parsing configuration to all peers prevents inconsistent event breaking, timestamp recognition, field extraction, or routing behavior. See Splunk's update configuration files in an indexer cluster documentation.

QUESTION NO: 17

A Splunk instance has the following settings in SPLUNK_HOME/etc/system/local/server.conf:

[clustering] mode = master replication_factor = 2 pass4SymmKey = password123

Which of the following statements describe this Splunk instance? (Select all that apply.)

A. This is a multi-site cluster.

B. This cluster's search factor is 2.

C. This Splunk instance needs to be restarted.

D. This instance is missing the master_uri attribute.

ANSWER: B C

Explanation:

The cluster's search factor is 2 because, when an indexer-cluster manager is configured without an explicit `search_factor` setting, Splunk Enterprise uses the default search factor of 2. The search factor defines how many searchable copies of bucket data the cluster maintains. The displayed configuration explicitly sets the replication factor to 2, while the absent search-factor setting therefore retains its default value. Splunk documents the default values for indexer clustering, including a replication factor of 3 and a search factor of 2 when those values are not otherwise configured.

This Splunk instance needs to be restarted because changes to the `[clustering]` stanza in `server.conf`, including enabling cluster-manager mode and defining clustering parameters, require a Splunk Enterprise restart to take effect. A restart causes the instance to initialize its clustering role using the configured replication setting and shared secret. The `pass4SymmKey` value is the shared cluster authentication key and must match the corresponding value used by participating cluster members. See Splunk's cluster manager configuration documentation and the server.conf specification for clustering settings and restart requirements.

QUESTION NO: 18

Which of the following statements describe licensing in a clustered Splunk deployment? (Select all that apply.)

A. Free licenses do not support clustering.

B. Replicated data does not count against licensing.

C. Each cluster member requires its own clustering license.

D. Cluster members must share the same license pool and license master.

ANSWER: A B D

Explanation:

Free licenses do not support clustering. Splunk Free is intended for a standalone deployment and does not provide the distributed-deployment capabilities required for clustered architectures. A clustered deployment must therefore use Splunk Enterprise licensing rather than a Free license.

Replicated data does not count against licensing. For an indexer cluster, Splunk calculates indexed-volume license usage from the original incoming data. The additional bucket copies created to satisfy replication and search-factor requirements are not counted again, preventing replication from artificially increasing the daily indexed-volume total.

Cluster members must share the same license pool and license master. Licensing in a distributed Splunk Enterprise deployment is centrally administered by a license manager (called a license master in older documentation). Cluster peers must be configured to use the same central licensing authority and license pool so that the deployment has a consistent view of its entitlement and indexed-volume usage. This central arrangement supports correct license enforcement across the cluster while ensuring replicated copies remain excluded from licensing volume. See Splunk's distributed deployment licensing documentation and its indexer cluster overview.

QUESTION NO: 19

When designing the number and size of indexes, which of the following considerations should be applied?

A. Expected daily ingest volume, access controls, number of concurrent users

B. Number of installed apps, expected daily ingest volume, data retention time policies

C. Data retention time policies, number of installed apps, access controls

D. Expected daily ingest volumes, data retention time policies, access controls

ANSWER: D

Explanation:

Expected daily ingest volumes, data retention time policies, access controls is correct because these factors directly determine an effective Splunk index design. Expected daily ingest volume is required for storage-capacity planning: it helps estimate the amount of raw and indexed data that will be generated, distributed, and retained over time. Retention policies determine how long data remains searchable and when aging data is frozen or removed, which in turn establishes the storage required for each index. Splunk supports retention settings such as `frozenTimePeriodInSecs` at the index level, making retention a core sizing consideration. Access controls also influence index boundaries because indexes can be used

with roles and index permissions to limit which populations may search particular datasets. Separating data based on meaningful access requirements supports security and can simplify administration. Together, ingestion rate, retention duration, and authorization needs provide the operational, capacity, and security inputs needed to determine both how many indexes are appropriate and how large they must be. Splunk's index-management guidance and indexes configuration reference describe these index-level storage, retention, and access-related design considerations. See About indexes and indexers and the indexes.conf specification.

QUESTION NO: 20

(A customer wishes to keep costs to a minimum, while still implementing Search Head Clustering (SHC). What are the minimum supported architecture standards?)

A. Three Search Heads and One SHC Deployer

B. Two Search Heads with the SHC Deployer being hosted on one of the Search Heads

C. Three Search Heads but using a Deployment Server instead of a SHC Deployer

D. Two Search Heads, with the SHC Deployer being on the Deployment Server

ANSWER: A

Explanation:

Three Search Heads and One SHC Deployer is the minimum supported architecture for a production Splunk search head cluster. Splunk requires at least three cluster members because the cluster relies on a majority quorum for captain election and for maintaining replicated cluster state. With three search heads, the cluster can continue to maintain quorum when one member is unavailable, which provides the minimum practical level of high availability while keeping infrastructure costs low. The captain coordinates activities such as scheduling and cluster-wide configuration management, so maintaining a majority of active members is essential to stable operation.

A deployer is also required operationally to distribute apps and configuration bundles to the search head cluster members. Administrators use the deployer to stage the appropriate bundle and push it consistently to all cluster members, rather than making configuration changes independently on each search head. The deployer is external to the cluster itself and is the supported mechanism for deploying cluster apps and other shared configurations.

For Splunk's search head clustering requirements and deployment guidance, see Search head cluster deployment overview and Use the deployer to distribute apps to a search head cluster.

QUESTION NO: 21

What is a Splunk Job? (Select all that apply.)

A. A user-defined Splunk capability.

B. Searches that are subjected to some usage quota.

C. A search process kicked off via a report or an alert.

D. A child OS process manifested from the splunkd process.

ANSWER: C

Explanation:

A search process kicked off via a report or an alert is a Splunk job. In Splunk Enterprise, a job represents the execution of a search: Splunk creates a search job when it runs a search and maintains job artifacts and status information while that search is active or retained. Scheduled knowledge objects, including reports and alerts, run their underlying searches on a schedule. Each such execution creates a search job, which can be inspected through the Jobs page or programmatically through Splunk's search job endpoints. Job properties commonly include the search identifier (SID), dispatch state, time range, runtime, event and result counts, and expiration information.

This concept is important for architecting search workloads because scheduled reports and alerts consume search-head resources as jobs. Administrators can monitor active and historical jobs, review execution behavior, and tune scheduling or workload management based on the job activity produced by these knowledge objects. Splunk's documentation describes search jobs as the entities used to execute and manage searches, while its reporting and alerting documentation explains that scheduled reports and alerts execute searches according to their configured schedules. See About search jobs and About alerts.

QUESTION NO: 22

To optimize the distribution of primary buckets; when does primary rebalancing automatically occur? (Select all that apply.)

A. Rolling restart completes.

B. Master node rejoins the cluster.

C. Captain joins or rejoins cluster.

D. A peer node joins or rejoins the cluster.

ANSWER: A B D

Explanation:

Primary rebalancing automatically occurs after a rolling restart completes, when the master node rejoins the cluster, and when a peer node joins or rejoins the cluster. In an indexer cluster, each replicated bucket has one designated primary copy. The primary copy is responsible for coordinating certain bucket-level activities, so an uneven primary-bucket distribution can create disproportionate workload on some peer nodes even when bucket copies themselves are evenly distributed.

These cluster lifecycle events can alter peer availability, membership, or the state of cluster coordination. Splunk therefore initiates automatic primary rebalancing to redistribute primary assignments across eligible peers and restore a more balanced workload. A completed rolling restart is handled as a cluster-wide event because each peer has been restarted in sequence; when the cluster returns to its normal state, Splunk can reassess the primary assignment distribution. Similarly, the return of the master node and peer membership changes provide opportunities for the cluster to rebalance primaries using the currently available peers.

In current Splunk terminology, the master node is called the cluster manager node. See Splunk's documentation on rebalancing an indexer cluster and the indexer cluster architecture.