celerating Your Data Transfo

FlashArray-Implementation-Specialist

Pure Storage Certified FlashArray Implementation Specialist

For More Information – Visit link below:

https://www.examsboost.com/

Product Version

- ✓ Up to Date products, reliable and verified.
- ✓ Questions and Answers in PDF Format.

Latest Version: 6.1

Question: 1

During a hardware NDU from FlashArray//XR2 or XR3 to an XR4 model, which default service onboard ports are NO longer present in the XR4 controller design?

A. iSCSI
B. Replication
C. Management

Answer: A

Explanation:
In the architectural evolution from FlashArray//XR2 and //XR3 to the FlashArray//XR4, Pure Storage made significant changes to the controller's rear panel and I/O design to accommodate higher performance and PCIe Gen 4 capabilities. One of the most critical changes for an Implementation Engineer to plan for during a hardware Non-Disruptive Upgrade (NDU) is the removal of the onboard, default iSCSI ports that were present on previous generations.
On //XR2 and //XR3 controllers, there were onboard Ethernet ports often used for iSCSI host connectivity without requiring a discrete PCIe adapter. However, the //XR4 controller design removes these dedicated onboard iSCSI ports to streamline the motherboard design and shift host I/O strictly to PCIe add-in cards. The //XR4 retains onboard ports specifically for Management (Mgmt) and Replication connectivity, ensuring that administrative access and array-to-array replication can still be maintained without additional hardware.
Consequently, if a customer was relying solely on the onboard ports for iSCSI traffic on their legacy //XR2 or //XR3, the Bill of Materials (BOM) for the upgrade to //XR4 must include discrete Ethernet/iSCSI PCIe cards to migrate those connections. Failure to identify this architectural difference during the pre-upgrade inventory can result in a loss of host connectivity or an inability to cable the new controllers correctly, as the expected physical ports simply will not exist on the new chassis.
=========

Question: 2

During a hardware NDU from a FlashArray//X20R3 to an X20R4-LL (low-line) model, the Implementation Engineer encounters a failure during the power supply check. Voltage readings are correct. What is a likely cause of this failure?

A. There are more than 10 drives installed in the chassis.
B. The power cables are NOT fully seated in the array.
C. Purity does NOT recognize the power supplies installed in the array.

Answer: A

Explanation:
The FlashArray//X20R4-LL (Low-Line) model is a specific configuration designed for lower capacity and power entry points. Unlike the standard //X or //XL models which can support fully populated chassis and shelves, the "Low-Line" configurations often come with strict hardware limitations regarding power draw and drive count. A known constraint for specific Low-Line chassis upgrades, particularly when moving to the X20R4-LL, is a limitation on the number of DirectFlash Modules (drives) supported due to the power supply unit (PSU) capacity or thermal design targeted for that specific SKU.
In this scenario, if the source array (X20R3) was populated with more drives than the target X20R4-LL supports—specifically more than 10 drives—the upgrade validation checks (such as those performed by the upgrade script or Purity health checks) will flag a failure. Even though the voltage readings are technically correct (indicating the PDUs are providing power), the check fails because the hardware configuration exceeds the supported power budget for that specific chassis model.
Implementation Engineers must verify the drive count of the source array against the specifications of the target "Low-Line" array during the planning phase. If the source array has 11 or more drives, the target hardware cannot be an LL model; it would require a standard model to support the additional power load of the extra drives. This check prevents the array from booting into a state where it might experience power contention or thermal shutdown under load.
=========

Question: 3

An Implementation Engineer attempts to add a FlashArray to a Fusion fleet but receives an error indicating the array cannot be added. The array appears healthy and is connected to Pure1. What is the most likely cause of this issue?

A. The array does NOT have any existing workloads configured.
B. The array is already assigned to another Fusion fleet.
C. The array is NOT running the latest Purity//FA version.

Answer: B

Explanation:
Pure Fusion is a storage-as-code platform that aggregates multiple physical FlashArrays into a single logical "fleet" or cloud-like storage pool. A fundamental rule of the Fusion architecture and registration process is that a specific physical FlashArray can only belong to one Fusion fleet at a time. This exclusivity ensures that the management plane (Fusion) has authoritative control over the resources, placement logic, and workload balancing for that array without conflict from another management entity.
When an Implementation Engineer encounters an error adding an array that is otherwise healthy and successfully connected to the Pure1 cloud (which acts as the mediator for Fusion), the most probable cause is that the array remains registered to a previous or different Fusion fleet. This often

happens if an array was used in a proof-of-concept (POC), a lab environment, or a previous deployment and was not cleanly deregistered before attempting to add it to the new production fleet.
To resolve this, the engineer must verify the array's registration status in Pure1 or the Fusion control plane. The array must be explicitly removed or deregistered from the conflicting fleet before it becomes available for claim by the new fleet. While Purity versions are important for compatibility, Fusion supports a range of versions, making "not running the latest" a less likely hard-stop error compared to the binary conflict of existing ownership. Similarly, existing workloads do not prevent an array from joining a fleet; Fusion can often import or manage existing resources depending on the configuration.
=========

Question: 4

After rebooting a controller, which command should an Implementation Engineer run to verify all the Purity services have started successfully?

A. pureadm
B. pureadm list
C. pureadm status --all

Answer: B

Explanation:
The pureadm command suite is the primary CLI tool used by Implementation Engineers and Support to manage the Purity operating environment's services and high-availability states. After a controller reboot—whether part of a hardware replacement, NDU, or fresh install—it is critical to verify that the Purity software stack has initialized correctly and that all sub-services (such as the I/O handling process, management daemon, and scut) are running.
The specific command pureadm list is the correct syntax to display a summary of the Purity services and their current status (e.g., running, stable, or stopped) on the local controller. This command provides a clean, immediate view of the service stack's health. If the output shows all expected services as "running," the engineer can confirm the controller has successfully rejoined the cluster and is ready to handle operations.
While pureadm alone might print help text and pureadm status is often a valid guess for other systems, the strict syntax for FlashArray implementation verification relies on pureadm list. This step is a standard part of the "Health Check" procedures found in upgrade guides. Failing to verify service status before proceeding (e.g., failing over to the other controller) could lead to an outage if the rebooted controller hasn't actually fully started its data services. Therefore, pureadm list is the validation checkpoint before declaring the controller healthy.
=========

Question: 5

Which PCI cards must be validated on the Bill of Materials (BOM) to allow an Implementation Engineer to attach a DirectFlash Shelf to a FlashArray//XL130R5?

A. FA-XL-200G Ethernet/RoCE 2-Port
B. FA-XL-100G iSCSI/RoCE 2-Port
C. FA-XL-100G Ethernet/RoCE 2-Port

Answer: C

Explanation:
The FlashArray//XL series (such as the XL130R5) utilizes a modern backend connectivity architecture for its expansion shelves, specifically the DirectFlash Shelf (DFS). Unlike older generations that might have used SAS (Serial Attached SCSI) for shelf connectivity, the //XL architecture relies on high-speed Ethernet with RoCE (RDMA over Converged Ethernet) to communicate with external media shelves. This ensures that the NVMe drives in the shelf perform with the same low latency as those in the head unit.
To attach a DirectFlash Shelf to a FlashArray//XL130R5, the chassis must be populated with specific PCIe interface cards designated for shelf back-end connectivity. The correct part for this is the FA-XL-100G Ethernet/RoCE 2-Port card. This card provides the necessary 100 Gigabit Ethernet bandwidth and RoCE protocol support required for the NVMe-oF (NVMe over Fabrics) backend fabric that Pure Storage uses.
It is crucial to distinguish this from host-facing cards. While an "iSCSI/RoCE" card exists, the shelf connectivity specifically utilizes the "Ethernet/RoCE" designation in the BOM to differentiate backend fabric ports from frontend host ports. The 200G options are generally reserved for host connectivity or inter-array clustering in specific high-performance setups, but the standard validated card for connecting the DFS to the //XL130R5 is the 100G Ethernet/RoCE adapter. Verifying this line item on the BOM prevents onsite delays where the engineer might otherwise find themselves with incompatible ports for the expansion shelves.