GPU SMITH
COMPUTE ENGINEERING

GPU SMITH / RESEARCH & PRACTICAL GUIDANCE

RoCE Fabric Acceptance Testing: PFC, ECN and CNP Runbook

> Plan and build private AI compute with GPU Smith. We connect workload requirements with GPU hardware, networking, deployment and operating decisions so your infrastructure fits the work it must perform.

Published by GPU Smith · 2026-09-19 · roce fabric acceptance testing · roce validation · priority flow control · explicit congestion notification · congestion notification packets · rdma networking · ethernet fabrics · gpu clusters · cumulus linux

Original article: https://gpusmith.com/articles/en/roce-fabric-acceptance-testing-runbook

In this article

Executive Summary

A **RoCE fabric acceptance test** should prove a bounded proposition: the exact hosts, network interface cards (NICs), links, switch application-specific integrated circuits (ASICs), network operating system (NOS) builds, firmware, and quality-of-service (QoS) policy under test behave as the approved design specifies. It cannot prove that every future workload or failure will be harmless. The evidence must therefore identify versions, topology, maximum transmission unit (MTU), trust mode, Differentiated Services Code Point (DSCP) or Priority Code Point (PCP), traffic-class mappings, Priority Flow Control (PFC), Explicit Congestion Notification (ECN), and Congestion Notification Packet (CNP) handling at every hop. NVIDIA's current documentation is pinned to **Cumulus Linux 5.18** (docs.nvidia.com), while its generated RoCE settings are ASIC-specific (docs.nvidia.com). These are examples for that release and platform, not universal settings.

The correct validation sequence is **freeze, baseline, load, correlate, restore**. First prove routing or VLAN reachability, end-to-end MTU, RoCEv2 classification on registered UDP port **4791** (iana.org), and consistent priority mapping. Then capture synchronized idle counters on every ingress, egress, and host. Apply a controlled ladder: single flow, message-size sweep, externally orchestrated many-to-one incast, bidirectional traffic, and the target collective. Perftest can report named sysfs counter deltas (github.com), while NCCL Tests checks both collective performance and correctness (github.com). Synthetic and collective tests answer different questions.

Interpret **deltas, direction, priority, and timing**, not isolated snapshots. ECN marks show a queue signaled congestion; RFC 3168 permits marking instead of dropping an ECN-capable packet (rfc-editor.org). The receiver then returns a CNP toward the sender, and the sender's reaction point should throttle. PFC is one-hop, priority-scoped backpressure, not end-to-end rate control. Linux exposes receive pause duration in microseconds per priority (docs.kernel.org). A rising switch no-buffer discard, NIC buffer discard, or host out-of-buffer counter is loss evidence, not a healthy congestion signal.

Acceptance requires buyer-supplied limits for **zero or permitted discards, pause duration, latency, throughput, and collective completion**. No defensible universal percentage exists. The evidence pack should contain raw before and after counters, timestamps, offered load, path, topology, configs, software inventory, artifacts, pass criteria, exceptions, and rollback state. GPU Smith's stated practice is to deliver engagements against written acceptance criteria and an as-built record (gpusmith.com); that adjacent engineering posture is useful here because the result should remain vendor-version-pinned and reproducible.

Introduction and Background

Remote Direct Memory Access over Converged Ethernet version 2 (**RoCEv2**) carries InfiniBand transport over User Datagram Protocol (UDP), allowing Layer 3 routing while the adapter performs the data movement. NVIDIA describes the encapsulation directly as InfiniBand transport over UDP (docs.nvidia.com). Fast data movement does not remove Ethernet congestion. Instead, a production design typically combines queue classification, ECN and endpoint rate response, with PFC only where the design calls for a lossless class.

The central acceptance question is not whether packets pass at idle. It is whether the intended data and feedback traffic remain correctly classified and observable while queues grow under representative contention. **ECN**, **CNP**, and **PFC** are different signals. ECN marks a packet at a congested queue. The receiver observes Congestion Experienced and sends a CNP back to the source. PFC tells a directly attached transmitter to stop one priority temporarily. IEEE defines PFC as per-traffic-class control on full-duplex links (1.ieee802.org), and identifies the class through VLAN priority (1.ieee802.org). PCP is a **three-bit** 802.1Q field (learn.microsoft.com).

This report is an operational **RoCE network validation checklist**, **RoCE PFC testing procedure**, **RoCE ECN validation**, and **RoCE CNP testing** guide. It also covers **RoCE buffer counter monitoring**, **PFC pause frame counters RoCE** operators need, a repeatable **data center RoCE acceptance test**, and evidence-based **RoCE congestion management testing**. The method is vendor-neutral at the decision level, but every actual command and expected counter must be pinned to the installed platform.

What a RoCE Acceptance Result Can Prove

An acceptance result proves only what its controlled boundary supports. Record the boundary before the first packet is generated. At minimum, the manifest should include:

- **Topology:** host, NIC port, cable, leaf, spine, peer leaf, destination port, and Equal-Cost Multipath (ECMP) path selection.
- **Switch identity:** manufacturer, model, ASIC generation, NOS image, boot image, active QoS profile, and configuration checksum.
- **Host identity:** operating system, kernel, RDMA core, driver, firmware, PCIe location, NUMA relationship, and active RoCE mode.
- **Packet contract:** VLAN, source and destination address, UDP destination port, MTU, DSCP, PCP, service class, traffic class, and queue. RFC 2474 requires codepoint-to-per-hop-behavior mappings to be configurable (rfc-editor.org).

- **Congestion contract:** lossless or lossy mode, ECN method, PFC priorities, CNP class and scheduling, headroom, shared-pool policy, and watchdog behavior.
- **Success contract:** offered load, test duration, message sizes, fan-in, permitted loss, latency distribution, throughput range, pause budget, and collective completion criteria.

Version capture is functional evidence, not paperwork. Upstream `ethtool -i` queries the driver associated with a network device (man7.org); NVIDIA documents `ofed_info -s` for the installed MLNX_OFED version (networking-docs.nvidia.com). Counter names are not portable: ethtool notes little commonality in vendor-specific statistic names (man7.org).

A passing run does **not** establish a universal ECN threshold, a generic safe PFC rate, or immunity to all traffic patterns. It does establish reproducible behavior for the recorded build. If firmware, cabling, MTU, topology, QoS trust, buffer policy, or test geometry changes, the affected tests should be rerun. NVIDIA's 5.18 release made dynamic ECN generally available (docs.nvidia.com), illustrating why release drift changes the tested system.

End-to-End Preflight and Configuration Freeze

Acceptance begins with a diffable, hop-by-hop matrix. NVIDIA's Ethernet Storage Fabrics guide says every network element in the path must handle RoCE (docs.nvidia.com). Microsoft similarly describes PFC as a link mechanism between directly connected peers (learn.microsoft.com). Data Center Bridging Exchange can expose conflicting NIC and link-partner QoS parameters (learn.microsoft.com), and carries its settings over LLDP (learn.microsoft.com). One correct leaf does not compensate for one incorrect host or spine.

Table 1 is the configuration consistency matrix to complete for every tested path. Values shown as “design” must come from the approved low-level design, bill of materials, or statement of work.

Field	Source host and NIC	Leaf ingress and egress	Spine	Peer leaf	Destination NIC
Version evidence	Kernel, RDMA core, driver, firmware, PSID	Model, ASIC, NOS, QoS profile	Model, ASIC, NOS, QoS profile	Model, ASIC, NOS, QoS profile	Kernel, RDMA core, driver, firmware, PSID
Reachability and MTU	VLAN or routed GID, IP MTU, RoCE mode	VLAN or VRF, routed MTU, port MTU	Routed MTU and ECMP state	VLAN or VRF, routed MTU, port MTU	Matching address family, GID, IP MTU
Classification	Application ToS, DSCP or PCP trust, user priority	Trust boundary, rewrite, DSCP/PCP to switch priority	Preserve or explicitly remark	Same mapping as source leaf	Matching trust, user priority, traffic class
RoCE data	Intended priority and hardware traffic class	Data classifier, queue, pool, scheduling	Same intended queue behavior	Data classifier, queue, pool, scheduling	Intended priority and receive queue
CNP feedback	Reaction-point counters and class	CNP classifier and scheduler	CNP class preserved	CNP classifier and scheduler	Notification-point marks and sent CNPs

Field	Source host and NIC	Leaf ingress and egress	Spine	Peer leaf	Destination NIC
PFC and ECN	PFC only on designed priorities; ECN enabled	ECN at egress; PFC at ingress in lossless design	Same mode and compatible thresholds	ECN at egress; PFC at ingress	Matching PFC and ECN behavior
Evidence file	`src-host-preflight.txt`	`leaf-a-preflight.txt`	`spine-preflight.txt`	`leaf-b-preflight.txt`	`dst-host-preflight.txt`

The table is deliberately symmetric. A mismatch in either direction can produce an asymmetric stall. Juniper's QFX procedure separates RoCE data and CNP classifiers (juniper.net), while Cisco NX-OS 10.5(x) requires the PFC-priority and QoS-group mapping to agree (cisco.com).

Run these **preflight checks** before generating congestion:

1. **Verify path and MTU:** test both directions and every ECMP path available to the harness. Confirm host, routed interface, and switch-port MTUs with non-fragmenting probes and RoCE traffic. The upstream `ip link set` interface is the documented mechanism to change a host device MTU (man7.org).
2. **Verify classification:** capture a packet at the source and at a reachable observation point. Confirm UDP 4791, DSCP, VLAN PCP where present, ECN-capable codepoint, and lack of unexpected remarking.
3. **Verify host mapping:** inspect application Type of Service, socket priority, user priority, hardware traffic class, and trust state. NVIDIA documents user-priority to traffic-class mapping through `mlnx_qos` or DCBX (networking-docs.nvidia.com). Its version-pinned interface can set trust to PCP or DSCP (networking-docs.nvidia.com).
4. **Verify switch mapping:** inspect classifier, switch priority, queue, service pool, scheduler, ECN policy, PFC state, and CNP class on every element.
5. **Verify persistence:** compare running and startup configurations. NVUE changes take effect after apply (docs.nvidia.com), and saving creates the startup configuration file (docs.nvidia.com).

Lossless and Lossy Congestion-Control Modes

Lossless RoCE usually means ECN plus PFC on the selected priority, with reserved or dynamically managed lossless headroom. **Lossy RoCE** means ECN-based feedback without PFC. The ESF guide explicitly permits ECN-only lossy mode (docs.nvidia.com). Cisco's blueprint says both hosts and network nodes must participate in the documented ECN and PFC design (cisco.com). Neither label guarantees correct operation. Acceptance must test the chosen mode, not assume one is intrinsically healthy.

The signaling sequence under congestion is:

- **Queue growth:** an egress queue crosses its configured marking condition.
- **ECN mark:** the switch sets Congestion Experienced on an ECN-capable packet rather than necessarily dropping it. ECN uses a two-bit IP field with four codepoints (rfc-editor.org).

- **Notification:** the destination NIC observes the mark and sends a CNP toward the original source. Cisco describes that receiver-generated feedback (cisco.com).
- **Reaction:** the source NIC's reaction point uses the CNP to reduce injection rate. A handled-CNP counter is therefore stronger evidence than a CNP merely arriving.
- **PFC backstop:** in a lossless design, the congested receiver can pause the selected priority on the adjacent link. PFC can propagate hop by hop, so direction and topology matter.

PFC is intentionally narrower than link-wide PAUSE. IEEE says it resembles 802.3x PAUSE but operates on individual priorities (1.ieee802.org). That selectivity still permits head-of-line blocking for unrelated flows sharing one queue. Peer-reviewed systems work identifies head-of-line blocking, congestion spreading, and occasional deadlocks as PFC limitations (arxiv.org). This is why **more pause is not automatically better**.

For a pinned example only, Cumulus Linux 5.18 documents switch priority **3** to traffic class **3** for RoCE and switch priority **6** to traffic class **6** for CNP (docs.nvidia.com). It documents an absolute ECN threshold of **1,501,500 bytes** for traffic class 3 (docs.nvidia.com). Those values are evidence about that documented release, not recommended acceptance limits for another ASIC, NOS, port speed, or topology.

Baseline Counters and Observability

Capture a synchronized baseline after the fabric is quiet but before counters are cleared. Keep both the original snapshot and the post-test snapshot. This preserves evidence if clearing is unsupported or itself changes state. Hardware-offloaded RoCE may not appear in ordinary Ethernet counters (networking-docs.nvidia.com), so collect RDMA and NIC hardware counters as well.

On Linux, `rdma statistic show` queries RDMA and driver-specific statistics (man7.org); with no object, it reports the default hardware counters (man7.org). Discover optional counters on the exact NIC and driver rather than assuming they exist.

Table 2 is a counter dictionary template. Replace every example name with the exact name and unit exposed by the installed build. “Project trigger” is deliberately not populated with a generic threshold.

Evidence	Location and example command	Unit or behavior	Expected qualitative response	Escalation trigger
RoCE bytes and packets	Switch: `nv show interface <id> qos roce counters`; host RDMA counters	Cumulative bytes and packets	Rise on traversed interfaces and intended priority	No rise, wrong port, or unexplained asymmetry
ECN marks	Switch egress ECN counter; NIC `cc_rx_ce_pkts` where supported	Packets	May rise during controlled queue growth	Marks with zero matching workload, or no marks when test intentionally crosses the approved condition
CNP sent, received, handled	Destination notification point, source receive and reaction point	Packets	CE reception, CNP transmission, then source handling should correlate	Missing link in the feedback chain or wrong priority

Evidence	Location and example command	Unit or behavior	Expected qualitative response	Escalation trigger
PFC pause frames	Switch per-priority Tx/Rx; NIC `rx_prio[p]_pause` and Tx equivalent	Packets	May rise on the congested priority in PFC mode	Pause on an unintended priority, direction mismatch, or project limit exceeded
PFC pause duration	Switch counter if unit documented; Linux mlx5 per-priority duration	Microseconds on documented mlx5 host counter	Delta aligns with congestion interval	Sustained pause after load ends or project pause budget exceeded
Buffer current and maximum	Switch ingress and egress pool or priority-group watermark	Bytes, cells, or percentage, platform specific	Maximum rises at the bottleneck, current value drains after load	Maximum reaches project limit or current use fails to drain
No-buffer discard	Switch RoCE no-buffer discard and NIC buffer-discard counters	Packets	Follows buyer-supplied loss criterion	Any disallowed positive delta
Host out of buffer	NIC hardware `out_of_buffer`	Events or packets, adapter specific	Normally unchanged unless receiver resources are exhausted	Positive delta under a test that requires zero host drops
Port errors and discards	Switch port and host link statistics	Frames or packets	No unrelated error growth	CRC, symbol, resource, or discard delta outside project criteria
Watchdog events	NOS PFC watchdog detected, recovered, or restored counters	Events	Unchanged in ordinary load; deterministic in approved injection	Event without injected cause, or failure to recover

The counter chain separates congestion from loss. Cumulus exposes a no-buffer discard for RoCE receive traffic (docs.nvidia.com); Linux mlx5 describes a receive-buffer discard as a packet discarded because buffers were unavailable (docs.kernel.org). NVIDIA's NIC `out_of_buffer` can instead indicate lack of work queue entries (networking-docs.nvidia.com). These counters point to different remediation domains.

Counter semantics also affect timing. Linux RDMA hardware reads can be cached, with a documented default lifespan of **10 milliseconds** (kernel.org). Arista clears watchdog counters when PFC or its watchdog is disabled (arista.com). Record collection intervals, clear operations, reloads, wraps, and configuration changes in the evidence log.

Controlled Load Ladder

Use increasing complexity so the first anomalous step narrows the search. Keep path, duration, warm-up, CPU and GPU affinity, message geometry, and process placement explicit. Perftest endpoints require matching mode-specific options (github.com). It can sweep message sizes from **2 bytes through 2^23 bytes** (github.com), measure bidirectional bandwidth (github.com), and save a report as JSON (github.com). Its maintainers state that the benchmarks do not emulate application traffic (github.com).

Table 3 defines the acceptance ladder and required raw artifacts. Durations, fan-in, and pass criteria are inputs, not defaults.

Step	Traffic pattern	Controlled variables	Expected signal	Raw artifact
0. Idle	No generated RoCE load	Same collection interval on every device	Stable baseline, no unexplained counter growth	`t00-idle-before.json`, `t00-idle-after.json`
1. Single flow	One sender to one receiver, each direction separately	Small, medium, and large messages; fixed QP count and duration	Correct path and classification, throughput and latency baseline	`t01-single-<path>-<size>.json`
2. Parallel flows	Multiple flows across planned ECMP paths	Same message and duration, increasing QPs or clients	Balanced path use, no unintended class change	`t02-parallel-<fanout>.json`
3. Incast	Multiple synchronized senders to one receiver	Explicit fan-in, aggregate offered load, receiver resources	ECN marks at bottleneck, CNP chain, bounded PFC if designed	`t03-incast-<fanin>-<load>.json`
4. Bidirectional	Simultaneous traffic in both directions	Same path pair and offered load per direction	Independent Tx and Rx results, no hidden asymmetry	`t04-bidir-<path>.json`
5. Collective	AllReduce, AllGather, ReduceScatter, or workload-required pattern	Rank placement, GPU and NIC affinity, size sweep, iterations	Correct completion and project-defined performance band	`t05-collective-<op>-<size>.log`
6. Approved injection	Link, process, or PFC-watchdog stimulus only in an isolated window	Change ticket, exact target, abort condition, restoration check	Defined detection and recovery, clean return to baseline	`t06-injection-<case>.tar`

The ladder prevents a collective from concealing a basic mapping error. Published RoCE research notes that topology-aware collectives can avoid incast and balance traffic (arxiv.org), so incast must be exercised separately. Conversely, NCCL Tests can span multiple processes, threads, and CUDA devices (github.com), which tests rank placement and collective behavior that perftest does not.

Operational rules for every step are:

- **Synchronize clocks and snapshots:** capture a common test ID and UTC timestamp on all hosts and switches.
- **Warm up consistently:** separate initialization from measured intervals and state the number of warm-up operations.
- **Change one axis at a time:** message size, QP count, fan-in, direction, or path, but not all simultaneously.
- **Preserve raw results:** save stdout, stderr, JSON where supported, counter snapshots, command lines, environment, and packet samples.
- **Stop on physical errors:** do not tune congestion control around new CRC, link, or symbol errors.
- **Limit injection:** execute failure or storm simulation only when approved, isolated, reversible, and paired with an abort condition. SONiC's own test plan verifies restoration after a simulated PFC storm (github.com).

Data Analysis and Evidence

The minimum useful analysis is a before and after delta joined to workload time. For each counter, compute **delta = after - before**, unless the device reports a gauge or watermark. A negative delta is not traffic reversal; it usually indicates a clear, wrap, reset, reload, or collection mistake that must be recorded. Juniper Apstra formalizes this idea as a time window over which interface-counter increments are measured (juniper.net).

Use these quantitative joins:

- **Offered load:** requested send rate, active senders, message bytes, QPs, and duration.
- **Delivered work:** transmitted and received RoCE bytes, completed operations, and collective correctness.
- **Congestion feedback:** egress ECN marks, receiver CE observations, CNPs sent, CNPs received, and CNPs handled.
- **Backpressure:** PFC Tx and Rx frames plus pause duration on the exact priority and adjacent links.
- **Queue pressure:** current use, maximum use, persistent watermark, and no-buffer discards.
- **End performance:** throughput, median and tail latency, completion time, and variance across repetitions.

The expected causal chain in an intentional congestion trial is **offered load, queue growth, ECN mark, receiver CNP, sender handling, rate response**. PFC may appear in a lossless design if the approved queue condition is reached, but it should be correlated hop by hop. Cisco describes PFC as traveling per hop from congestion toward the source (cisco.com). Dell's Enterprise SONiC material exposes a command for PFC statistics across Ethernet interfaces (infohub.delltechnologies.com), and its RoCE checks include persistent headroom watermarks (infohub.delltechnologies.com).

Retain the complete achieved-bandwidth series, not only an aggregate. RFC 6349 recommends logged output across the full test duration (datatracker.ietf.org).

BDP (Hypothetical Example)

Bandwidth-delay product (**BDP**) is a first-pass occupancy calculation, not a switch threshold. RFC 6349 defines BDP in bits as round-trip time multiplied by bottleneck bandwidth (datatracker.ietf.org) and divides by eight for bytes (datatracker.ietf.org). For an illustrative **400 Gb/s** bottleneck and measured **20 microsecond** round trip:

`400,000,000,000 bits/s x 0.000020 s / 8 = 1,000,000 bytes`, or ~0.954 MiB.

That number excludes implementation margins, in-flight frame completion, MAC and PHY delay, cable distance, packet-size distribution, pipeline delay, port speed transitions, multiple ingress contributors, cell rounding, shared-pool policy, and vendor safety factors. NVIDIA notes that PFC headroom includes PHY and MAC delay (docs.nvidia.com). Use BDP only to make assumptions explicit, then use the validated ASIC and NOS design for actual buffer settings.

Troubleshooting by Counter Deltas

Troubleshooting starts at the first broken link in the evidence chain. Avoid interpreting any single counter without path, priority, direction, and time.

- **Throughput collapses, ECN marks rise, CNPs are handled, no discards:** rate control is active. Compare achieved throughput and latency with the project band, then examine whether marking begins at the intended queue condition.
- **ECN marks rise, destination CNP sent stays flat:** verify ECN-capable bits, destination notification-point counters, NIC firmware, and CNP generation. The mark reached a queue, but feedback did not complete.
- **CNP sent rises, source CNP received or handled stays flat:** trace reverse-path classification, CNP traffic class, scheduling, and source reaction-point configuration.
- **PFC Tx rises on a switch, adjacent NIC or switch PFC Rx stays flat:** check physical adjacency, priority mapping, counter direction, telemetry interval, and whether the inspected interface is the actual path.
- **NIC receive-pause rises:** Linux mlx5 documentation interprets this as the network being unable to absorb traffic from that adapter (docs.kernel.org). Follow the path toward the congested egress.
- **NIC transmit-pause rises:** the adapter is signaling that it cannot absorb incoming traffic. Linux describes that state as the adapter being unable to absorb traffic (docs.kernel.org). Inspect receiver queues, work queue entries, host scheduling, and application posting behavior.
- **No-buffer discard rises before meaningful ECN or CNP:** suspect classification, an ECN policy that is absent or too late for the validated design, an unobserved microburst, or insufficient buffer. Do not simply increase thresholds.
- **Pause appears on the wrong priority:** inspect DSCP or PCP trust and rewrite boundaries. DSCP selects per-hop behavior at each node (rfc-editor.org); a mapping mismatch can put data or CNP into the wrong queue.
- **Buffer maximum is high but current use drains:** a transient burst occurred. Arista describes watermarks as exposing bursty, transient ingress use (arista.com). Correlate its timestamp with offered load and pause.
- **Current buffer use or pause persists after load:** stop further escalation, verify the generator ended, inspect watchdog and queue state, and follow the approved recovery plan.
- **Only one direction fails:** compare route, ECMP member, MTU, trust, CNP reverse path, and receiver resources independently. RFC 6349 recommends measuring directions separately before simultaneous bidirectional testing (datatracker.ietf.org).
- **Collective stalls but perftest passes:** compare rank-to-NIC affinity, actual collective paths, message sizes, process placement, and control-plane connectivity. A point-to-point pass does not exercise the collective geometry.

This sequence separates a switch congestion condition from a receiver resource problem. Junos OS Evolved 24.4R1 exposes PFC and ECN monitoring through telemetry sensors (juniper.net). Cisco NX-OS 10.4(x) can show real-time and peak shared-buffer state per port (cisco.com). Use each platform's native view to reconstruct the same causal chain.

Acceptance Evidence Pack and Rollback

The acceptance pack is the product of the test. A dashboard screenshot is not sufficient because it cannot reproduce filters, intervals, resets, or raw values. Store immutable raw files plus a human-readable summary.

Required contents are:

- **Scope record:** purchase order or statement of work reference, design revision, test boundary, exclusions, and named acceptance criteria.
- **Topology record:** physical and logical diagrams, cabling, port maps, ECMP paths, rail mapping, and traffic-generator placement.
- **Version record:** switch model and ASIC, NOS and boot image, NIC part and PSID, firmware, driver, kernel, RDMA core, CUDA and collective versions.
- **Configuration record:** running and startup configs, QoS objects, trust and rewrite rules, PFC and ECN state, CNP class, queue scheduling, buffer policy, and checksums.
- **Test manifest:** command line, environment, source and destination, path, direction, message sizes, QPs, ranks, fan-in, duration, warm-up, repetitions, and offered load.
- **Raw evidence:** before and after counters, deltas, telemetry time series, packet captures, workload outputs, logs, and timestamps.
- **Decision record:** measured result, criterion, pass or fail, exception owner, remediation, retest reference, and sign-off.
- **Rollback record:** pre-test state, every temporary change, clear or reset action, restoration proof, and post-test quiet baseline.

Do not clear counters without recording it. Cumulus has an interface-specific RoCE clear action (docs.nvidia.com), but no portable host-side reset semantics were found across the examined stacks. The safe cross-platform method is synchronized snapshots and explicit deltas. Where counters wrap or reset, invalidate that interval and repeat it.

Rollback means more than removing the traffic generator. Confirm that temporary QoS changes are gone, test VLANs or routes are removed if required, watchdog or telemetry state is restored, buffers have drained, links are stable, and counters remain quiet for the project-defined observation window. GPU Smith describes integration and acceptance testing against written criteria (gpusmith.com); as an adjacent advisor, it belongs in this evidence and decision process, not in a vendor feature comparison.

Implications and Future Directions

RoCE acceptance is moving from static configuration inspection toward **closed-loop observability**. A mature system should join switch queue telemetry, NIC congestion-control counters, workload timestamps, and topology in one time base. The immediate benefit is not a prettier dashboard. It is the ability to distinguish marking, feedback, backpressure, and loss in the order they occurred.

Three design implications follow:

- **Telemetry is part of acceptance:** if ECN marks, CNP reception and handling, pause direction, buffer maxima, and discards cannot be observed on the installed versions, the fabric is not yet supportable enough for production.
- **Unused bandwidth may remain shareable:** IEEE's Enhanced Transmission Selection allows other traffic classes to use available bandwidth (1.ieee802.org). Acceptance should therefore record background traffic as well as the RoCE class.
- **Version control is part of performance:** driver, firmware, NOS, QoS profile, and counter schema must be managed together. DOCA notes that its exact counter list depends on driver version (networking-docs.nvidia.com).
- **Thresholds are design outputs:** link speed, topology, cable distance, MTU, packet-size distribution, ASIC buffer architecture, and endpoint response all matter. NVIDIA warns that an incorrect cable-length setting can waste buffer (docs.nvidia.com).

Dynamic and telemetry-assisted congestion controls will change exact commands and perhaps reduce reliance on PFC, but the acceptance logic remains durable: verify classification, create controlled pressure, observe the complete feedback loop, prove the absence or boundedness of loss and pause against supplied criteria, then preserve the evidence. ECN itself does not guarantee zero loss. RFC 3168 states that a full queue can still force a packet drop (rfc-editor.org).

Frequently Asked Questions (FAQs)

What is the shortest useful RoCE validation checklist?

Freeze versions and topology, prove MTU and classification end to end, capture idle switch and NIC counters, run a single flow in each direction, add incast and bidirectional pressure, run the target collective, calculate deltas, compare them with buyer-supplied criteria, and restore the original state. The checklist is incomplete if it omits raw artifacts or CNP handling.

Should PFC pause counters be zero?

Not universally. In an ECN-only design, PFC should not be part of the data path. In a PFC-enabled design, pause may occur under the approved pressure case, but its priority, direction, duration, propagation, and post-load recovery must satisfy the project criteria. IEEE's objective is to prevent congestion-related frame loss (1.ieee802.org), not to maximize pause traffic.

How is RoCE ECN validation different from CNP testing?

ECN validation proves that the intended queue marks ECN-capable packets under a controlled condition. CNP testing proves the destination recognizes those marks, sends feedback on the correct class, and the source receives and handles it. NVIDIA's `rp_cnp_handled` definition ties handling to transmission throttling (networking-docs.nvidia.com).

What constitutes zero-loss evidence?

It is the absence of disallowed positive deltas in every relevant switch discard, NIC buffer-discard, host out-of-buffer, and physical-error counter for the exact test interval, combined with complete workload results. It is not a single switch counter at zero, and it does not prove all future workloads are lossless.

When should an acceptance team change ECN or PFC thresholds?

Only when the validated design identifies the platform, release, reason, target value, and rollback. A failed test should first identify classification, visibility, endpoint response, or resource problems. It should not trigger trial-and-error threshold changes. The current Cumulus guidance recommends leaving dynamic settings to the platform unless the operator understands the lossless requirements (docs.nvidia.com).

Conclusion

A defensible RoCE fabric acceptance result is a synchronized body of evidence, not a throughput screenshot. It begins with a frozen manifest, proves packet classification and feedback across every hop, observes idle state, and applies a load ladder that separates point-to-point transport, incast, bidirectional behavior, and collective traffic. The analysis then joins workload time to ECN, CNP, PFC, buffer, discard, and host-resource deltas.

The most important discipline is to preserve distinctions. An ECN mark is congestion signaling, a handled CNP is endpoint response, a PFC frame is one-hop backpressure, and a discard is loss. High buffer use is not itself a failure, while a zero snapshot is not itself proof. Direction, priority, timing, version, and topology give each counter meaning.

Finally, the test must use the acceptance limits supplied by the validated design or statement of work. BDP arithmetic can expose assumptions, and documented vendor defaults can explain one pinned release, but neither is a universal threshold. A fabric is ready for production when it meets the agreed performance and loss criteria, produces a coherent counter chain under controlled load, remains observable enough to diagnose deviations, and returns cleanly to its recorded baseline.

Source links

Links cited in this article, in order of first appearance. Full addresses are provided for readers using extracted text.

1. https://docs.nvidia.com/networking-ethernet-software/cumulus-linux/Whats-New/
2. https://docs.nvidia.com/networking-ethernet-software/cumulus-linux/Layer-1-and-Switch-Ports/Quality-of-Service/RDMA-over-Converged-Ethernet-RoCE/
3. https://www.iana.org/assignments/service-names-port-numbers?search=IP+Routable+RocE

4. https://github.com/linux-rdma/perftest/blob/master/man/perftest.1
5. https://github.com/nvidia/nccl-tests
6. https://www.rfc-editor.org/info/rfc3168/
7. https://docs.kernel.org/6.10/networking/device_drivers/ethernet/mellanox/mlx5/counters.html
8. https://gpusmith.com/
9. https://1.ieee802.org/dcb/802-1qbb/
10. https://learn.microsoft.com/en-us/windows-hardware/drivers/network/overview-of-data-center-bridging
11. https://www.rfc-editor.org/info/rfc2474/
12. https://man7.org/linux/man-pages/man8/ethtool.8.html
13. https://networking-docs.nvidia.com/mlnxofedswum/23106161lts/rdma-over-converged-ethernet-roce
14. https://docs.nvidia.com/networking-ethernet-software/guides/esf-generic-config-guide/
15. https://www.juniper.net/documentation/us/en/software/platform/qfx5220-qfx5230-qfx5240-qfx5241-qfx5250-configuration-essentials/qfx-configuration-essentials/topics/task/ROCEv2.html
16. https://www.cisco.com/c/en/us/td/docs/dcn/nx-os/nexus9000/105x/configuration/qos/cisco-nexus-9000-series-nx-os-quality-of-service-configuration-guide-105x/m-configuring-priority-flow-control.html
17. https://man7.org/linux/man-pages/man8/ip-link.8.html
18. https://networking-docs.nvidia.com/mlnxofedswum/23106161lts/quality-of-service-qos
19. https://www.cisco.com/c/en/us/td/docs/dcn/whitepapers/cisco-data-center-networking-blueprint-for-ai-ml-applications.html
20. https://arxiv.org/abs/1806.08159
21. https://www.man7.org/linux/man-pages/man8/rdma-statistic.8.html
22. https://networking-docs.nvidia.com/doca/archive/3-5-0/doca-telemetry-service-guide
23. https://www.kernel.org/doc/html/latest/admin-guide/abi-stable.html
24. https://www.arista.com/en/um-eos/eos-dcbx-and-flow-control?searchword=eos+section+32+1+introduction
25. https://github.com/linux-rdma/perftest/blob/master/README
26. https://arxiv.org/abs/2207.10898
27. https://github.com/sonic-net/SONiC/wiki/PFC-Watchdog-Test-Plan
28. https://www.juniper.net/documentation/us/en/software/apstra6.0/apstra-user-guide/topics/concept/probe-interface-queue-stats-monitoring.html
29. https://infohub.delltechnologies.com/en-us/l/dell-technologies-enterprise-sonic-with-azure-local/queue-counters-1/
30. https://infohub.delltechnologies.com/en-us/l/ai-fabrics-nvidia-spectrum-with-dell-sonic-proof-of-concept/roce-system-checks-2/4/
31. https://datatracker.ietf.org/doc/html/rfc6349
32. https://docs.nvidia.com/networking-ethernet-software/cumulus-linux/Layer-1-and-Switch-Ports/Quality-of-Service/
33. https://www.arista.com/en/support/toi/tag/pfc
34. https://www.juniper.net/documentation/us/en/software/junos/release-notes/24.4/junos-evo-release-notes-24.4r1/topics/new-features/feature-descriptions/junos-telemetry-interface-3.html
35. https://www.cisco.com/c/en/us/td/docs/dcn/nx-os/nexus9000/104x/configuration/qos/cisco-nexus-9000-series-nx-os-quality-of-service-configuration-guide-104x/m-configuring-queuing-and-scheduling.html

36. https://docs.nvidia.com/networking-ethernet-software/nvue-reference/Action-Commands/

37. https://1.ieee802.org/dcb/802-1qaz/

THE PUBLISHER

About GPU Smith

GPU Smith provides independent engineering and research for private AI infrastructure. We help technical buyers and operators reason about GPU workload sizing, hardware selection, procurement, deployment and inference operations, from the compute system to the networking, power and cooling requirements around it.

Start with the workload

Useful infrastructure decisions begin with the models, data, throughput, latency and operating constraints a team actually has. GPU Smith helps connect those requirements with system design and deployment choices. The goal is a privately controlled AI environment whose capacity and operational demands are understood before a purchasing decision.

Hardware and supplier research

Our public reference library covers GPUs, complete systems and networking components. The server-vendor directory and manufacturing research help teams investigate suppliers, compare options and follow sources behind technical claims. We distinguish manufacturer specifications from measured benchmarks and advertised capabilities from tested configurations. Reference pages are research resources, not a live inventory listing or a binding equipment quote.

Deployment and operations

GPU Smith's engineering scope includes infrastructure integration, networking, power, cooling and the practical operation of inference workloads. Our published articles explain the tradeoffs behind deployment, procurement and ongoing operations so teams can ask better questions and document decisions.

Work with GPU Smith

Explore the hardware library, GPU references, networking references and vendor research. Contact GPU Smith with your workloads and deployment constraints to discuss a project and confirm current scope, pricing and availability.

Inclusion of a manufacturer or product in our research does not imply a partnership, certification or endorsement.

Website: https://gpusmith.com