Switch Abstraction Interface (SAI) Release Notes v2211.24.0.0

Table of Contents

Release Notes Update History 3
Overview 4
Prerequisites 4
Changes and new Features 5
SAI General Support 6
Supported Switch Systems 6
SDK Compatibility 6
Bug Fixes in this Version 8
Known Issues 9
Unsupported or Not Implemented Functions / Attributes 14
Release Notes History 15
Changes and new Features History 15
Bug Fixes History 15

Release Notes Update History

Revision	Date	Description
2211.24.0.0	January 31, 2023	Initial release of this Release Notes version.

Overview

SAI, Switch Abstraction Interface, defines the API to provide a hardware-independent layer of controlling Switch ASIC in a uniform manner. SAI is part of the Open Compute Project (OCP). SAI is an open-source project managed by the OCP in https://github.com/opencomputeproject/SAI

> ⚠ Not all the APIs defined for SAI v1.11.0 are fully implemented in this release.

Prerequisites

As a precondition to running SAI initialize switch:

- SDK must not be running (If it is running, unload the SDK process and sx_core driver. If you are working on a DVS system, you can use as a shortcut `dvs_stop.sh`).
- The application must provide one key/value pair on profile services, depending on system type:

```
"config_file=/usr/share/sai_2700.xml" /
"config_file=/usr/share/sai_2410.xml" /
"config_file=/usr/share/sai_2420.xml" /
"config_file=/usr/share/sai_2100.xml" /
"config_file=/usr/share/sai_2010.xml" /
"config_file=/usr/share/sai_2201.xml" /
"config_file=/usr/share/sai_3420.xml" /
"config_file=/usr/share/sai_3700.xml" /
"config_file=/usr/share/sai_4410.xml" /
"config_file=/usr/share/sai_4600.xml" /
"config_file=/usr/share/sai_4600C.xml" /
"config_file=/usr/share/sai_4700.xml" /
"config_file=/usr/share/sai_5600.xml"
```

Upon shutting down the switch, SAI will perform all the necessary clean up and terminate the SDK.

Changes and new Features

Feature/Change	Description
Rev. 2211.24.0.0	
ACL	Added support for an ACL match on the AETH field (`SAI_ACL_TABLE_ATTR_FIELD_AETH_SYNDROME`, `SAI_ACL_ENTRY_ATTR_FIELD_AETH_SYNDROME`) to count RoCE NAK and CNP packets.
PLL Status	Added a new logging entry that alerts the user upon a PLL lock loss event.
Dual ToR: Additional MAC Address	Added support for setting a MAC address for the router interface which is not part of the 10 bit MAC address available for RIFs on Spectrum-1, as part of the Dual ToR scenario.
Dual ToR: DSCP Remapping	Added support for tunnel QoS maps as part of the Dual TOR scenario.
mlxtrace	Added mlxtrace output to the SAI debug dump to show techsupport.
NOS Response for SDK Failure	Enabled SDK sysfs health mechanism and reporting health event monitoring in case of failure.
Bug Fixes	See Bug Fixes.

SAI General Support

Supported Switch Systems

This software supports the switch systems listed below.
SAI v2211.24.0.0 is tested on x86 and Arm architectures for the list of switches specified below.

Part Number	System Description
SN4700	NVIDIA Spectrum-3 based 400GbE, 1U Open Ethernet switch, 32xQSFP-DD ports, x86 CPU, standard depth
SN4600V	NVIDIA Spectrum-3 based 200GbE 2U Open Ethernet switch, 64 QSFP56 ports, 2 Power Supplies (AC), x86 CPU, standard depth
SN4600C	NVIDIA Spectrum-3 based 100GbE 2U Open Ethernet Switch, 64 QSFP56 ports, 2 Power Supplies (AC), standard depth
SN4410	NVIDIA Spectrum-3 based 400GbE 1U Open Ethernet switch, 24 QSFPDD28 and 8 QSFP-DD ports, 2 Power Supplies (AC), x86 CPU, standard depth
SN3700C	NVIDIA Spectrum-2 based 100GbE Open Ethernet switch, 32 QSFP28 ports, x86 CPU, standard depth
SN3700	NVIDIA Spectrum-2 based 200GbE Open Ethernet switch, 32 QSFP56 ports, x86 CPU, standard depth
SN3420	NVIDIA Spectrum-2 based 25GbE/100GbE 1U Open Ethernet switch, 48 SFP28 ports and 12 QSFP28 ports, x86 CPU, Standard depth
SN2700	NVIDIA Spectrum based 100GbE, Open Ethernet switch, 32 QSFP28 ports, x86 CPU, standard depth
SN2410	NVIDIA Spectrum based 25GbE/100GbE, Open Ethernet switch, 48 SFP28 ports, 8 QSFP28 ports, x86 dual core, short depth
SN2201	NVIDIA Spectrum based 1GBase-T/100GbE 1U Open Ethernet switch with ONIE, 48 RJ45 ports and 4 QSFP28 ports, Dual Power Supply (AC), x86 CPU, short depth, C2P airflow, 4-post Rail kit
SN2100	NVIDIA Spectrum based 100GbE, Open Ethernet switch, 16 QSFP28 ports, short depth, Rangeley CPU
SN2010	NVIDIA Spectrum based 100GbE, Open Ethernet switch, 18 SFP+ and 4 QSFP28 ports, 2 AC power supplies, x86 quad core, short depth

SDK Compatibility

SAI is always tested with compatible SDK and Firmware versions.

Switch Components	Version	Additional Information
NVIDIA Spectrum-3 Firmware	30.2000.5006	
NVIDIA Spectrum-2 Firmware	29.2000.5006	
NVIDIA Spectrum Firmware	13.2000.5006	
SDK	4.5.5008	SDK API can be found at: https://github.com/Mellanox/SwitchRouterSDK-interfaces

Switch Components	Version	Additional Information
SAI API	1.11.0	

Bug Fixes in this Version

This section provides only a list of bugs fixed by NVIDIA® only in this version.

RM #	Issue
3312610	**Description:** When setting a WRED profile attribute for a color that was not enabled during the profile create time, an error would be returned. After the fix, a default profile is create on such scenario and the set attribute is applied on top of it
	Keywords: WRED
	Discovered in Version: SAI 2205.23.1.0
	Fixed in Version: SAI 2211.24.0.0
3281919	**Description:** When calling the flush FDB by using the `SAI_FDB_FLUSH_ATTR_BRIDGE_PORT_ID` attribute, the bridge `bv_id` value was filled on the notification callback where it should have been left empty.
	Keywords: FDB
	Discovered in Version: SAI 2205.23.1.0
	Fixed in Version: SAI 2211.24.0.0
3255889	**Description:** `SAI_NATIVE_HASH_FIELD_IP_PROTOCOL` only affected hashing of IPv4 traffic, (thru the hashing IPv4 Protocol field), it did not affect the IPv6 traffic's hashing. After the fix, it also affect hashing of IPv6 traffic, by hashing IPv6 next header field.
	Keywords: HASH, IPv4/IPv6
	Discovered in Version: SAI 2205.23.1.0
	Fixed in Version: SAI 2211.24.0.0

Known Issues

Component/Feature	Description

ACL	• Not all match keys / actions are supported
	• VLAN ranges are not implemented
	• Supports 16 ACL ranges objects (8 when ISSU is enabled)
	• ACL switch bind points SAI_SWITCH_ATTR_DEFAULT_INGRESS_ACL_LIST, SAI_SWITCH_ATTR_DEFAULT_EGRESS_ACL_LIST are not implemented
	• ACL table can only be a member of a single sequential ACL group
	• ACL table can only be a member in multiple groups if all these groups are parallel
	• ACL VLAN bind point only works for ingress stage, SAI_VLAN_ATTR_EGRESS_ACL is not supported
	• VLAN matching on egress RIF or egress Port is not supported for bridge interfaces (802.1D). It is supported only for VLAN interfaces (802.1Q) and for Port-vlan interfaces
	• VLAN matching is not supported when there is no RIF attached to the port
	• ACL should be applied on LAG and not on member ports of a LAG
	• ACL egress block list action can be applied only on egress stage, not on ingress stage
	• ACL field out port does not support LAG
	• ACL field in port does not support LAG member ports (Spectrum1, 2, 3)
	• ACL IP_TYPE field supports the following values:
	  • SAI_ACL_IP_TYPE_ANY (When using SAI_ACL_IP_TYPE_ANY, it is necessary to set at list one other field, as internally no key is used to match ANY value, and at least one key should be set for every rule)
	  • SAI_ACL_IP_TYPE_ARP
	  • SAI_ACL_IP_TYPE_IP
	  • SAI_ACL_IP_TYPE_IPV4ANY
	  • SAI_ACL_IP_TYPE_IPV6ANY
	  • SAI_ACL_IP_TYPE_NON_IP
	• ACL SAI_ACL_ENTRY_ATTR_ACTION_MIRROR_INGRESS is not supported at Egress stage. Note an exception, for SAI_ACL_BIND_POINT_TYPE_ROUTER_INTERFACE at Egress stage, only SAI_ACL_ENTRY_ATTR_ACTION_MIRROR_INGRESS can be used.
	• SAI_ACL_ENTRY_ATTR_ACTION_ADD_VLAN_PRI is supported only when paired with SAI_ACL_ENTRY_ATTR_ACTION_ADD_VLAN_ID as actions of the same ACL entry
	• When setting next hop group as a target of ACL action redirect, traffic will be redirected only to the next hop group members that are not tunnel encap next hops
	• SAI_ACL_ENTRY_ATTR_FIELD_TC supports only boolean mask, either exact match or match all

IP-over-IP	• Set is not supported on any attribute in sai_tunnel_attr_t
	• SAI_TUNNEL_ATTR_DECAP_MAPPERS cannot be created or is able to set
	• Only SAI_TUNNEL_TTL_PIPE_MODEL is supported for SAI_TUNNEL_ATTR_DECAP_TTL_MODE
	• Only SAI_TUNNEL_IPINIP and SAI_TUNNEL_IPINIP_GRE are supported for SAI_TUNNEL_ATTR_TYPE and SAI_TUNNEL_TERM_TABLE_ENTRY_ATTR_TUNNEL_TYPE on create
	• get_tunnel_attribute function will fail when trying to get the following values from a tunnel created without any encap attributes
	  • SAI_TUNNEL_ATTR_ENCAP_DSCP_MODE
	  • SAI_TUNNEL_ATTR_ENCAP_DSCP_VAL
	• Currently DSCP mode, DSCP value and ECN mode are shared between the same type of tunnels. Change DSCP mode, DSCP value and ECN mode will affect all the tunnels of the same type. Tunnels are considered 'the same type' if both the following conditions are satisfied:
	  a. SAI_TUNNEL_ATTR_TYPE has the same value
	  b. Tunnel direction has the same value. Direction has 3 types: encap, decap and symmetry
	  i. Encap: no decap attributes (start with SAI_TUNNEL_ATTR_ENCAP_) are configured during tunnel creation
	  ii. Decap: no encap attributes (start with SAI_TUNNEL_ATTR_DECAP_) are configured during tunnel creation
	  iii. Symmetry: Both encap and decap attributes are configured during tunnel creation
	• Only drop is supported for SAI_TUNNEL_ATTR_LOOPBACK_PACKET_ACTION

VXLAN	• Only SAI_TUNNEL_TTL_PIPE_MODEL is supported for SAI_TUNNEL_ATTR_ENCAP_TTL_MODE on Spectrum-1. Spectrum-2 and onwards also support SAI_TUNNEL_TTL_MODE_UNIFORM_MODEL
	• When creating VXLAN tunnel without SAI_TUNNEL_ATTR_ENCAP_SRC_IP, it is not possible to create tunnel map entries before tunnel term entries are created
	• SAI_VLAN_ATTR_UNKNOWN_UNICAST_FLOOD_CONTROL_TYPE, SAI_VLAN_ATTR_BROADCAST_FLOOD_CONTROL_TYPE should be the same for specific VLAN as far as remote endpoints configuration.
	  • If either UC or BC has a flood group which contains remote endpoints, and the second traffic type (BC or UC respectively) has a configuration without remote endpoints, flooding for both UC and BC to remote endpoints will occur according to the first traffic type.
	  • If both UC and BC have a flood group which contains remote endpoints, and the remote endpoints differ, the configuration done later setting takes effect.
	
	The same applies on SAI_BRIDGE_ATTR_UNKNOWN_UNICAST_FLOOD_CONTROL_TYPE , SAI_BRIDGE_ATTR_BROADCAST_FLOOD_CONTROL_TYPE for a specific bridge.

Flood Vectors	• SAI_VLAN_ATTR_UNKNOWN_UNICAST_FLOOD_CONTROL_TYPE and SAI_VLAN_ATTR_BROADCAST_FLOOD_CONTROL_TYPE should be the same for specific VLAN as far as remote endpoints configuration.
	  • If either UC or BC has a flood group which contains remote endpoints, and the second traffic type (BC or UC respectively) has a configuration without remote endpoints, flooding for both UC and BC to remote endpoints will occur according to the first traffic type.
	  • If both UC and BC have a flood group which contains remote endpoints, and the remote endpoints differ, the configuration done later setting takes effect. Same applies for SAI_BRIDGE_ATTR_UNKNOWN_UNICAST_FLOOD_CONTROL_TYPE, SAI_BRIDGE_ATTR_BROADCAST_FLOOD_CONTROL_TYPE for specific bridge

WRED	• In case WRED profile has WRED enabled for at least one color, and ECN mark mode enabled for at least one color, the enabled colors for WRED and ECN must be the same (For example, cannot have SAI_WRED_ATTR_GREEN_ENABLE=true and SAI_WRED_ATTR_ECN_MARK_MODE=SAI_ECN_MARK_MODE_YELLOW)
Queue Stats	• Clearing queue stats for specific counters always clears all available counters • Clearing SAI_QUEUE_STAT_WRED_ECN_MARKED_PACKETS for a single queue ECN marked packets clears all the port queues ECN marked packets
Host Interface	• Host interface user-defined-traps support a single instance of each user defined trap. • Host interface wildcard-table-entry overwrites the configured channel value for all traps at the time of the creation, there is no priority implemented. To avoid such an issue, first create the wildcard entry and then specific entries.

UDF	• UDF Match:
	  • Only L2 matches which can be ARP/IPv4/IPv6 ethertype or empty are supported
	  • Empty match is only supported for SAI_UDF_BASE_L2
	  • Non-empty matches (ARP/IPv4/IPv6) are only supported for SAI_UDF_BASE_L3
	• UDF Base:
	  • If L2 – Support only one UDF in the UDF group
	  • If L3 – Support up to 3 UDFs in the UDF group. UDF matches should be unique within the UDF group
	• UDF Group - Empty UDF group cannot be used as hash or ACL attribute. In addition, it is not allowed to remove the last UDF from the UDF group if it is in use
	• ECMP hash distribution is inferior when using UDF with hash type CRC.
	Workaround: When using UDF for ECMP hash, use hash type XOR.

Mirroring / SPAN	• When using all available mirror sessions (3 Spectrum-1, 8 Spectrum-2 onwards), it is not possible to edit a session. As a workaround, remove and re-create the session with new values. It is possible to edit a session when using less than maximum available sessions (1 or 2 on Spectrum-1, 1-7 on Spectrum-2+) • When working with WRED discards mirroring, it has to be configured last after regular mirror sessions, as with WRED discard mirroring all ports which are not analyzer ports of any session, are set as monitor port, and analyzer and monitor behavior for port is mutually exclusive • Mirror policer should be single-rate two color (mode = STORM_CONTROL or mode = SR_TCM with PBS = 0) • On Spectrum-1, mirror policer only works for a mirror session that is used as SAI_ACL_ENTRY_ATTR_ACTION_MIRROR_INGRESS or SAI_ACL_ENTRY_ATTR_ACTION_MIRROR_EGRESS. It does not affect other traffic that is mirrored by the session from non-ACL binding (such as port binding) • Mirror sample rate is supported only on Spectrum-2 onwards, and only works for a mirror session that is used as SAI_ACL_ENTRY_ATTR_ACTION_MIRROR_INGRESS or SAI_ACL_ENTRY_ATTR_ACTION_MIRROR_EGRESS. It does not affect other traffic that is mirrored by the session from non-ACL binding (such as port binding) • For packets matching ACL entry with mirror action and mirror policer, the color will be set to green after the ACL • When working with router discards mirroring, it is not possible to use WJH tool with extended discard traps
QoS Maps	• Switch Prio (SAI TC) <-> IEEE PFC is a global property, therefore, when setting different PFC->PG on different ports last value overrides
Debug Counters	• Debug counters do not work correctly with WJH, and vice-versa, as the 2 features configurations conflicts • A specific drop reason can only exist in a single counter • When enabling ACL drops debug counter, the dropped packets will not be counted by SDK ingress/egress policy engine discard counters
ISSU / Hitless warmboot	• ISSU 1 second is not supported with SDK 4.3.23** and newer • ISSU hitless does not support empty LAG with no members • It is not possible to create bridge port over empty LAG during ISSU. To avoid this issue, create at least one LAG member, then create bridge port over the LAG
Resource Monitoring	• SAI_SWITCH_ATTR_AVAILABLE_IPV6_ROUTE_ENTRY measures only IPv6 routes with prefix < 64 on Spectrum-1 and free space on single KVD hash. • SAI_ACL_TABLE_ATTR_AVAILABLE_ACL_ENTRY is not accurate on Spectrum-2.
Counters	• Host interface trapped packet/bytes values are cleared whenever setting SAI_HOSTIF_TRAP_ATTR_COUNTER_ID • When setting nexthop group, and next hop group members counter, the later one set "wins" • Get counter attribute for nexthop group member returns last applied counter - both group and member

LAG	All the members of a LAG need to have the same sFlow packet sampling rate. If the rate for one of the member port is updated, the change will be reflected on all the member ports. **Workaround:** Maintain the same sampling rate on all the member ports of a LAG.
Generic	Hash algorithm CRC CCITT is not supported on NVIDIA Spectrum-1, it is supported on NVIDIA Spectrum-2 onwards.

Unsupported or Not Implemented Functions / Attributes

The following functions / attributes are not supported / implemented:

- SAI_NEXT_HOP_ATTR_ROUTER_INTERFACE_ID get in router interface API
- SAI_PORT_ATTR_FDB_LEARNING_LIMIT_VIOLATION in port API
- Port advertised and remote advertised attributes
- Only the mandatory set of attributes are accepted on Create port
- IP/IPv6 counters, SAI_PORT_STAT_IF_IN_VLAN_DISCARDS, SAI_PORT_STAT_IF_OUT_QLEN, SAI_PORT_STAT_ETHER_RX_OVERSIZE_PKTS, SAI_PORT_STAT_ETHER_TX_OVERSIZE_PKTS, SAI_PORT_STAT_ETHER_STATS_TX_NO_ERRORS, SAI_PORT_STAT_ETHER_STATS_RX_NO_ERRORS, Port EEE counters, Port PFC ON2OFF counters, clear port stats for specific counters, in port counters API
- Port ether stats/in/out packets 4K-9K counters count only packets 4K-8K
- SAI_ROUTE_ATTR_NEXT_HOP_GROUP_ID get in route API
- SAI_VIRTUAL_ROUTER_ATTR_SRC_MAC_ADDRESS
- SAI_VIRTUAL_ROUTER_ATTR_VIOLATION_TTL1_ACTION, SAI_VIRTUAL_ROUTER_ATTR_VIOLATION_IP_OPTIONS in router API
- SAI_SWITCH_ATTR_FDB_TABLE_SIZE, SAI_SWITCH_ATTR_BCAST_CPU_FLOOD_ENABLE, SAI_SWITCH_ATTR_MCAST_CPU_FLOOD_ENABLE, SAI_SWITCH_ATTR_VIOLATION_TTL1_ACTION get, SAI_SWITCH_ATTR_SRC_MAC_ADDRESS set, SAI_SWITCH_ATTR_MAX_LEARNED_ADDRESSES, SAI_SWITCH_ATTR_FDB_MULTICAST_MISS_ACTION set in switch API
- SAI_BRIDGE_ATTR_MAX_LEARNED_ADDRESSES is implemented for .1D bridges but not for the .1Q bridge
- VLAN counters API
- Notification callbacks - switch shutdown request
- saiobject.h query objects functions
- Some bulk functions
- Sample packet is supported only on ingress traffic to port, at profile mode (exclusive instance per port). Netflow mode (shared, aggregated traffic), Egress traffic on port (SAI_PORT_ATTR_EGRESS_SAMPLEPACKET_ENABLE) and ACL ingress/egress (SAI_ACL_ENTRY_ATTR_ACTION_INGRESS_SAMPLEPACKET_ENABLE, SAI_ACL_ENTRY_ATTR_ACTION_EGRESS_SAMPLEPACKET_ENABLE) are not supported.

Release Notes History

Changes and new Features History

Feature/Change	Description
Rev. 2205.23.1.0	
Bug Fixes	See Bug Fixes
Rev. 2205.23.0.0	
Point-to-Point (P2P) Tunnel Peer Mode	Added P2P tunnel peer mode for IP-in-IP tunnels.
Error Correction Code (ECC) Logging and Reporting	Added support for logging and reporting ECC errors, including the counter values.
Per-port IP Statistics	Added per port IP statistics (`SAI_PORT_STAT_IP_*`, `SAI_PORT_STAT_IPV6_*`). **Note:** This feature is controlled by the system XML profile node `per-port-ip-counter-enabled`, and is not enabled by default.
Bug Fixes	See Bug Fixes

Bug Fixes History

This section provides only a list of bugs fixed by NVIDIA® only in this version.

RM #	Issue
3242505	**Description:** Hash inner source IP and inner destination IP could not be set as hash fields for general ECMP and LAG hash configuration and ECMP IPv4 and LAG IPv4 traffic hash configuration. This version provides a fix to this issue and now these fields can be selected for all types of hash configuration
	Keywords: Hash inner source IP and inner destination IP
	Discovered in Version: SAI 2205.23.0.0
	Fixed in Version: SAI 2205.23.1.0
3241471	**Description:** When the user did not provide `SAI_TUNNEL_ATTR_ENCAP_SRC_IP` on IP-in-IP tunnel creation, the tunnel creation operation failed. This version provides a fix to this issue and now the default value of 0.0.0.0 is correctly used as the source IP.
	Keywords: Tunnel creation, IP-on-IP
	Discovered in Version: SAI 2205.23.0.0
	Fixed in Version: SAI 2205.23.1.0

RM #	Issue
3240476	**Description:** When the user did not provide SAI_TUNNEL_ATTR_DECAP_DSCP_MODE on on IP-in-IP tunnel creation, the tunnel creation operation failed. This version provides a fix to this issue and now when the value is not provided, the default value of uniform DSCP mode is used instead.
	Keywords: Tunnel creation, IP-on-IP
	Discovered in Version: SAI 2205.23.0.0
	Fixed in Version: SAI 2205.23.1.0
3262015	**Description:** When creating a SAI dump, sai_dbg_generate_dump() would return to caller while the last extended SDK/firmware debug dump is being written. In this case, the user would see a temporary file ending with .tmp with content changing This version provides a fix to this issue and now SAI waits until the file processing is done before returning to caller.
	Keywords: SAI dump
	Discovered in Version: SAI 2205.23.0.0
	Fixed in Version: SAI 2205.23.1.0
3016257	**Description:** An error would be returned when trying to add a port as a new LAG member if the LAG is a member in any VLAN for which the unknown unicast or broadcast flood control is set to none This version provides a fix to this issue.
	Keywords: LAG
	Discovered in Version: SAI 2205.23.0.0
	Fixed in Version: SAI 2205.23.1.0

RM #	Issue
3131450	**Description:** When deleting an L2MC group from the `SAI_VLAN_ATTR_UNKNOWN_UNICAST_FLOOD_GROUP` and then trying to delete it from the `SAI_VLAN_ATTR_BROADCAST_FLOOD_GROUP` (meaning, first broadcast then unicast), the user would get an error. Issue is also relevant for `SAI_BRIDGE_ATTR_UNKNOWN_UNICAST_FLOOD_GROUP`, `SAI_BRIDGE_ATTR_BROADCAST_FLOOD_GROUP`.
	Keywords: VXLAN
	Discovered in Version: SAI 1.22.0.1
	Fixed in Version: SAI 2205.23.0.0
3118367	**Description:** Applying a scheduler group/subgroup to port **after** adding the port as a LAG member is not functional. This version provides a fix to this issue.
	Keywords: LAG
	Discovered in Version: SAI 1.21.2.0
	Fixed in Version: SAI 2205.23.0.0
3131451	**Description:** On some cases, L2 learning operation was not disabled on VXLAN tunnels, but the remote entries were learned. Learning is now disabled.
	Keywords: VXLAN

RM #	Issue
	Discovered in Version: SAI 1.22.0.1
	Fixed in Version: SAI 2205.23.0.0
3139303	**Description:** On some cases, when the ACL group is bound as ingress/egress to a port, and port is removed from a LAG, the ACL group did not re-bound to the port. This version provides a fix to this issue.
	Keywords: ACL
	Discovered in Version: SAI 1.21.2.0
	Fixed in Version: SAI 2205.23.0.0
3141872	**Description:** Moving the FDB entry from the local bridge port to a remote bridge port failed. This version fixed the bad flow.
	Keywords: VXLAN
	Discovered in Version: SAI 1.22.0.1
	Fixed in Version: SAI 2205.23.0.0
3177465	**Description:** When trying to remove a QoS map that was removed from a port while it was a LAG member, an error that the QoS map is in use would be returned. This version provides a fix that enables QoS map removing.
	Keywords: QoS mapping
	Discovered in Version: SAI 1.22.0.1
	Fixed in Version: SAI 2205.23.0.0
3177465	**Description:** On NVIDIA Spectrum-2/NVIDIA Spectrum-3 switch systems, when a policer is bound to a mirror session, it must be single rate and color blind, and red action drop. If any of its attributes is different, SAI will internally override them (to maintain backwards compatibility with previous versions).
	Keywords: Policer, mirror sessions
	Discovered in Version: SAI 1.22.0.1
	Fixed in Version: SAI 2205.23.0.0

Notice

This document is provided for information purposes only and shall not be regarded as a warranty of a certain functionality, condition, or quality of a product. Neither NVIDIA Corporation nor any of its direct or indirect subsidiaries and affiliates (collectively: “NVIDIA”) make any representations or warranties, expressed or implied, as to the accuracy or completeness of the information contained in this document and assumes no responsibility for any errors contained herein. NVIDIA shall have no liability for the consequences or use of such information or for any infringement of patents or other rights of third parties that may result from its use. This document is not a commitment to develop, release, or deliver any Material (defined below), code, or functionality.

NVIDIA reserves the right to make corrections, modifications, enhancements, improvements, and any other changes to this document, at any time without notice.

Customer should obtain the latest relevant information before placing orders and should verify that such information is current and complete.

NVIDIA products are sold subject to the NVIDIA standard terms and conditions of sale supplied at the time of order acknowledgement, unless otherwise agreed in an individual sales agreement signed by authorized representatives of NVIDIA and customer (“Terms of Sale”). NVIDIA hereby expressly objects to applying any customer general terms and conditions with regards to the purchase of the NVIDIA product referenced in this document. No contractual obligations are formed either directly or indirectly by this document.

NVIDIA products are not designed, authorized, or warranted to be suitable for use in medical, military, aircraft, space, or life support equipment, nor in applications where failure or malfunction of the NVIDIA product can reasonably be expected to result in personal injury, death, or property or environmental damage. NVIDIA accepts no liability for inclusion and/or use of NVIDIA products in such equipment or applications and therefore such inclusion and/or use is at customer’s own risk.

NVIDIA makes no representation or warranty that products based on this document will be suitable for any specified use. Testing of all parameters of each product is not necessarily performed by NVIDIA. It is customer’s sole responsibility to evaluate and determine the applicability of any information contained in this document, ensure the product is suitable and fit for the application planned by customer, and perform the necessary testing for the application in order to avoid a default of the application or the product. Weaknesses in customer’s product designs may affect the quality and reliability of the NVIDIA product and may result in additional or different conditions and/or requirements beyond those contained in this document. NVIDIA accepts no liability related to any default, damage, costs, or problem which may be based on or attributable to: (i) the use of the NVIDIA product in any manner that is contrary to this document or (ii) customer product designs.

No license, either expressed or implied, is granted under any NVIDIA patent right, copyright, or other NVIDIA intellectual property right under this document. Information published by NVIDIA regarding third-party products or services does not constitute a license from NVIDIA to use such products or services or a warranty or endorsement thereof. Use of such information may require a license from a third party under the patents or other intellectual property rights of the third party, or a license from NVIDIA under the patents or other intellectual property rights of NVIDIA.

Reproduction of information in this document is permissible only if approved in advance by NVIDIA in writing, reproduced without alteration and in full compliance with all applicable export laws and regulations, and accompanied by all associated conditions, limitations, and notices.

THIS DOCUMENT AND ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, “MATERIALS”) ARE BEING PROVIDED “AS IS.” NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. TO THE EXTENT NOT PROHIBITED BY LAW, IN NO EVENT WILL NVIDIA BE LIABLE FOR ANY DAMAGES, INCLUDING WITHOUT LIMITATION ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE, OR CONSEQUENTIAL DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, ARISING OUT OF ANY USE OF THIS DOCUMENT, EVEN IF NVIDIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Notwithstanding any damages that customer might incur for any reason whatsoever, NVIDIA’s aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms of Sale for the product.

Trademarks

NVIDIA, the NVIDIA logo, and Mellanox are trademarks and/or registered trademarks of NVIDIA Corporation and/or Mellanox Technologies Ltd. in the U.S. and in other countries. Other company and product names may be trademarks of the respective companies with which they are associated.