ZebOS-XP Static Route SMI Reference

IP Infusion Inc.

Generated by Doxygen 1.6.1

Wed Dec 16 12:33:54 2015

Contents

Chapter 1

File Index

1.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 2

File Documentation

2.1 smi_rib.h File Reference

Provides APIs for managing static_routes. `#include "smi_client.h"`
`#include "smi_rib_msg.h"`

Functions

- int **smi_rib_multipath_num_func_sdkapi_validate** (struct smiclient_globals ∗azg, int set, int mutipathNum)
- int **smi_rib_set_maximum_static_routes_sdkapi_validate** (struct smiclient_globals ∗azg, int vrId, int num)
- int **smi_rib_unset_maximum_static_routes_sdkapi_validate** (struct smiclient_globals ∗azg, int vrId)
- int **smi_rib_set_maximum_fib_routes_sdkapi_validate** (struct smiclient_globals ∗azg, int vrId, int num)
- int **smi_rib_unset_maximum_fib_routes_sdkapi_validate** (struct smiclient_globals ∗azg, int vrId)
- int **smi_rib_fib_retain_set_validate** (struct smiclient_globals ∗azg, u_int32_t vrId, int retainTime)
- int **smi_rib_fib_retain_unset_validate** (struct smiclient_globals ∗azg, u_int32_t vrId)
- int **smi_rib_ipv4_route_stale_clear_validate** (struct smiclient_globals ∗azg, u_int32_t vrId)
- int **smi_rib_ipv6_route_stale_clear_validate** (struct smiclient_globals ∗azg, u_int32_t vrId)
- int **smi_rib_ip_route_ifprefix_set_sdkapi_validate** (struct smiclient_globals ∗azg, int vrId, char ∗ipv4DestinationPrefix, char ∗ifname, int distanceValue, u_int32_t tagValue, char ∗staticRouteDescription)
- int **smi_rib_ip_route_ifprefix_unset_sdkapi_validate** (struct smiclient_globals ∗azg, int vrId, char ∗ipv4DestinationPrefix, char ∗ifname, int distanceValue, u_int32_t tagValue, char ∗staticRouteDescription)

- int **smi_rib_ip_route_prefix_set_sdkapi_validate** (struct smiclient_globals ∗azg, int vrId, char ∗ipv4DestinationPrefix, char ∗ipv4GatewayStr, int distanceValue, u_int32_t tagValue, char ∗staticRouteDescription)
- int **smi_rib_ip_route_prefix_unset_sdkapi_validate** (struct smiclient_globals ∗azg, int vrId, char ∗ipv4DestinationPrefix, char ∗ipv4GatewayStr, int distanceValue, u_int32_t tagValue, char ∗staticRouteDescription)
- int **smi_rib_ip_route_unset_sdkapi_validate** (struct smiclient_globals ∗azg, int vrId, struct prefix_ipv4 ∗ipv4)
- int **smi_rib_ip_route_all_vrf_unset_sdkapi_validate** (struct smiclient_globals ∗azg, int vrId, struct prefix_ipv4 ∗ipv4)
- int **smi_rib_ipv4_route_vrf_ifname_set_sdkapi_validate** (struct smiclient_globals ∗azg, int vrId, char ∗vrfName, struct prefix_ipv4 ∗ipv4Prefix, struct pal_in4_addr ∗ipv4GateAddr, char ∗ifName, int distance, int metric, u_int32_t tag, char ∗desc)
- int **smi_rib_ipv4_route_vrf_ifname_unset_sdkapi_validate** (struct smiclient_globals ∗azg, int vrId, char ∗vrfName, struct prefix_ipv4 ∗ipv4Prefix, struct pal_in4_addr ipv4GateAddr, char ∗ifName, int distance, u_int32_t tag, char ∗desc)
- int **smi_rib_ip_mroute_prefix_set_sdkapi_validate** (struct smiclient_globals ∗azg, int vrId, char ∗vrfName, char ∗mrouteIpv4Prefix, char ∗mrouteGateStr, char ∗mrouteifname, char ∗routeType, int mrouteDistance)
- int **smi_rib_ip_mroute_prefix_unset_sdkapi_validate** (struct smiclient_globals ∗azg, int vr_id, char ∗vrfName, char ∗ipv4_prefix, char ∗route_type)
- int smi_rib_multipath_num_func_sdkapi (struct smiclient_globals ∗azg, int set, int mutipathNum)

  Set multipath numbers installed to FIB.
- int smi_rib_set_maximum_static_routes_sdkapi (struct smiclient_globals ∗azg, int vrId, int num)

  Set maximum static routes number.
- int smi_rib_unset_maximum_static_routes_sdkapi (struct smiclient_globals ∗azg, int vrId)

  Set maximum static routes number to the default value 4294967294UL.
- int smi_rib_set_maximum_fib_routes_sdkapi (struct smiclient_globals ∗azg, int vrId, int num)

  Set maximum fib routes number. Allowed number of fib routes excluding Kernel, Connect and Static.
- int smi_rib_unset_maximum_fib_routes_sdkapi (struct smiclient_globals ∗azg, int vrId)

  Set maximum fib routes number to the default value 4294967294.
- int smi_rib_fib_retain_set (struct smiclient_globals ∗azg, u_int32_t vrId, int retainTime)

  Retain FIB for a specific time after RIB restarts.

- int smi_rib_fib_retain_unset (struct smiclient_globals ∗azg, u_int32_t vrId)

    Retain FIB retain time to default value.

- int smi_rib_ipv4_route_stale_clear (struct smiclient_globals ∗azg, u_int32_t vrId)

    Function to clear IPv4 stale kernel routes from NSM RIB and FIB.

- int smi_rib_ipv6_route_stale_clear (struct smiclient_globals ∗azg, u_int32_t vrId)

    Function to clear IPv6 stale kernel routes from NSM RIB and FIB.

- int smi_rib_ip_route_prefix_set_sdkapi (struct smiclient_globals ∗azg, int vr_id, char ∗ipv4_prefix, char ∗gate_str, int distance, u_int32_t tag, char ∗desc)

    Function to configure ipv4 static route.

- int **smi_rib_ip_route_ifprefix_set_sdkapi** (struct smiclient_globals ∗azg, int vr_id, char ∗ipv4_prefix, char ∗ifname, int distance, u_int32_t tag, char ∗desc)
- int smi_rib_ip_route_prefix_unset_sdkapi (struct smiclient_globals ∗azg, int vr_id, char ∗ipv4_prefix, char ∗gate_str, int distance, u_int32_t tag, char ∗desc)

    Function to clear configured ipv4 static route.

- int **smi_rib_ip_route_ifprefix_unset_sdkapi** (struct smiclient_globals ∗azg, int vr_id, char ∗ipv4_prefix, char ∗ifname, int distance, u_int32_t tag, char ∗desc)
- int smi_rib_ipv4_route_set_sdkapi (struct smiclient_globals ∗azg, int vrId, char ∗vrfName, struct prefix_ipv4 ∗ipv4Prefix, struct pal_in4_addr ∗ipv4GateAddr, char ∗ifname, int distance, int metric, int snmpRouteType, u_int32_t tag, char ∗desc)

    Establish ipv4 static routes into VRF.

- int smi_rib_ipv4_route_unset_sdkapi (struct smiclient_globals ∗azg, int vrId, char ∗vrfName, struct prefix_ipv4 ∗ipv4Prefix, struct pal_in4_addr ∗ipv4GateAddr, char ∗ifname, int distance, u_int32_t tag, char ∗desc)

    clear ipv4 static routes into VRF

- int smi_rib_ipv4_route_vrf_ifname_set_sdkapi (struct smiclient_globals ∗azg, int vrId, char ∗vrfName, struct prefix_ipv4 ∗ipv4Prefix, struct pal_in4_addr ∗ipv4GateAddr, char ∗ifName, int distance, int metric, u_int32_t tag, char ∗desc)

    Establish ipv4 static routes into VRF interfacename.

- int smi_rib_ipv4_route_vrf_ifname_unset_sdkapi (struct smiclient_globals ∗azg, int vrId, char ∗vrfName, struct prefix_ipv4 ∗ipv4Prefix, struct pal_in4_addr ipv4GateAddr, char ∗ifName, int distance, u_int32_t tag, char ∗desc)

    clear ipv4 static routes into VRF interface name

- int **smi_rib_ip_mroute_prefix_set_sdkapi** (struct smiclient_globals ∗azg, int vrId, char ∗vrfName, char ∗mrouteIpv4Prefix, char ∗mrouteGateStr, char ∗mrouteifname, char ∗routeType, int mrouteDistance)
- int **smi_rib_ip_mroute_prefix_unset_sdkapi** (struct smiclient_globals ∗azg, int vr_id, char ∗vrfName, char ∗ipv4_prefix, char ∗route_type)
- int smi_rib_show_route_ipv6_sdkapi (struct smiclient_globals ∗azg, int startIndex, int endIndex, int vrId, char ∗vrfName, int database, u_char type, struct list ∗ribList, int(∗callback)(struct list ∗ribList))

  Function to establish IPV6 static routes into VRF.

- int smi_rib_show_route_ipv4_sdkapi (struct smiclient_globals ∗azg, int startIndex, int endIndex, int vrId, char ∗vrfName, int database, u_char type, struct list ∗ribList, int(∗callback)(struct list ∗ribList))

  Use this function to get all ipv4 routes from the routing table.

- int smi_rib_show_ipv4_route_nhaddr (struct smiclient_globals ∗azg, int startIndex, int endIndex, char ∗vrfName, int vrId, char ∗ipAddr, struct list ∗ribList, int(∗callback)(struct list ∗ribList))

  Use this function to get all ipv4 routes from the routing table for specified next hop address.

- int smi_rib_show_ipv6_route_nhaddr (struct smiclient_globals ∗azg, int startIndex, int endIndex, char ∗vrfName, int vrId, char ∗ipAddr, struct list ∗ribList, int(∗callback)(struct list ∗ribList))

  Use this function to get all ipv6 routes from the routing table for specified next hop address.

- int smi_rib_show_ipv6_route_interface (struct smiclient_globals ∗azg, int startIndex, int endIndex, char ∗vrfName, int vrId, char ∗ifName, struct list ∗ribList, int(∗callback)(struct list ∗ribList))

  Use this function to get all ipv6 routes from the routing table for specified interface name.

- int smi_rib_show_ipv4_route_interface (struct smiclient_globals ∗azg, int startIndex, int endIndex, char ∗vrfName, int vrId, char ∗ifName, struct list ∗ribList, int(∗callback)(struct list ∗ribList))

  Use this function to get all routes from the routing table for specified interface name.

- int smi_rib_get_vrf_details (struct smiclient_globals ∗azg, int startIndex, int endIndex, char ∗vrfName, int vrId, struct list ∗vrfList, int(∗callback)(struct list ∗vrfList))

  Use this function to get all VRF details.

- int smi_rib_show_ipv6_route_details_sdkapi (struct smiclient_globals ∗azg, int startIndex, int endIndex, char ∗vrfName, int vrId, char ∗ipAddr, struct list ∗ribList, int(∗callback)(struct list ∗ribList))

  Use this function to get ipv6 routing table for the specified ip address.

- int smi_rib_show_ipv4_route_details_sdkapi (struct smiclient_globals ∗azg, int startIndex, int endIndex, char ∗vrfName, int vrId, char ∗ipAddr, struct list ∗ribList, int(∗callback)(struct list ∗ribList))

  Use this function to get ipv4 routing table for the specified ip address.

- int smi_rib_show_route_summary_ipv4 (struct smiclient_globals ∗azg, char ∗vrfName, int vrId, struct smi_route_summ ∗rtSumm)

  Function to get summary of all ipv4 routes.

- int smi_rib_show_route_summary_ipv6 (struct smiclient_globals ∗azg, char ∗vrfName, int vrId, struct smi_route_summ ∗rtSumm)

  Function to get summary of all ipv6 routes.

- int smi_rib_get_vrf_given_gateway_sdkapi (struct smiclient_globals ∗azg, int vrId, char ∗str, vrf_id_t vrfId, int family)

  Get the vrfId given gateway(Gateway can be ifName or nh address).

- int smi_rib_ip_route_unset_sdkapi (struct smiclient_globals ∗azg, int vrId, char ∗vrfName, struct prefix_ipv4 ∗ipv4)

  Function to clear configured ipv4 static route.

- int smi_rib_ip_route_all_vrf_unset_sdkapi (struct smiclient_globals ∗azg, int vrId, char ∗vrfName, struct prefix_ipv4 ∗ipv4)

  Function to clear configured ipv4 static route.

- int smi_rib_debug (struct smiclient_globals ∗azg, int vrId, int debug)

  Use this function to specify the options for the displayed debugging information for RIP events, RIP packets and RIP NSM.

- int smi_rib_no_debug (struct smiclient_globals ∗azg, int vrId, int debug)

  Use this function to disable specific debugging.

- int **smi_rib_clear_ip_route_sdkapi** (struct smiclient_globals ∗azg, afi_t afi, int vrId, char ∗vrfName, char ∗prefixStr)

2.1.1 Detailed Description

Provides APIs for managing static_routes.

2.1.2 Function Documentation

2.1.2.1 int smi_rib_debug (struct smiclient_globals ∗ *azg*, int *vrId*, int *debug*)

Use this function to specify the options for the displayed debugging information for RIP events, RIP packets and RIP NSM. smi_rib_debug

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router Id

← ***debug*** Pass debug flag as following:
SMI_RIB_DBG_ALL - Debug all RIB information
SMI_RIB_DBG_EVENTS - Debug RIB events
SMI_RIB_DBG_PACKET - Debug RIB and NSM communications
SMI_RIB_DBG_PACKET_SEND - Debug sent packets
SMI_RIB_DBG_PACKET_RECV - Debug received packets
SMI_RIB_DBG_PACKET_DETAIL - Display detailed information for the sent and received packet
SMI_RIB_DBG_PACKET_SEND_DETAIL - Display detailed information for the sent packet
SMI_RIB_DBG_PACKET_RECV_DETAIL - Display detailed information for the received packet

Returns:

0 on success, otherwise one of the following error codes RIB_API_SET_ERR_VR_NOT_EXIST

2.1.2.2 int smi_rib_fib_retain_set (struct smiclient_globals ∗ *azg*, u_int32_t *vrId*, int *retainTime*)

Retain FIB for a specific time after RIB restarts. smi_rib_fib_retain_set

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router ID

← ***retainTime*** Retain time range is <0-65535>. O means means FIB forever. In other case Specific time should be mentioned.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.3 int smi_rib_fib_retain_unset (struct smiclient_globals ∗ *azg*, u_int32_t *vrId*)

Retain FIB retain time to default value. smi_rib_fib_retain_unset

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router ID

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.4 int smi_rib_get_vrf_details (struct smiclient_globals ∗ *azg*, int *startIndex*, int *endIndex*, char ∗ *vrfName*, int *vrId*, struct list ∗ *vrfList*, int(∗)(struct list ∗vrfList) *callback*)

Use this function to get all VRF details. smi_rib_get_vrf_details

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***startIndex*** start index

← ***endIndex*** end index

← ***vrfName*** VRF Name. Pass NULL to get all VRF details in a list.

← ***vrId*** Virtual Router Id

→ ***vrfList*** Link list of structure smi_vrf_details. smi_vrf_details structure holds details of single VRF. List should be intialized by caller.

→ ***callback*** Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_vrf_details. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.5 int smi_rib_get_vrf_given_gateway_sdkapi (struct smiclient_globals ∗ *azg*, int *vrId*, char ∗ *str*, vrf_id_t *vrfId*, int *family*)

Get the vrfId given gateway(Gateway can be ifName or nh address). smi_rib_get_vrf_given_gateway_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** virtual id [in] str Gateway string. (ifName or nh_adress) [out] vrfId Updated vrfId if found else not updated. [in] family AF_INET or AF_INET6

Returns:

RIB_API_SET_SUCCESS on success,otherwise one of the following error codes
RIB_API_SET_ERR_VR_NOT_EXIST

2.1.2.6 int smi_rib_ip_route_all_vrf_unset_sdkapi (struct smiclient_globals ∗ *azg*, int *vrId*, char ∗ *vrfName*, struct prefix_ipv4 ∗ *ipv4*)

Function to clear configured ipv4 static route. smi_rib_ip_route_all_vrf_unset_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router Id

← ***ipv4*** Ipv4 Address

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.7 int smi_rib_ip_route_prefix_set_sdkapi (struct smiclient_globals ∗ *azg*, int *vr_id*, char ∗ *ipv4_prefix*, char ∗ *gate_str*, int *distance*, u_int32_t *tag*, char ∗ *desc*)

Function to configure ipv4 static route. smi_rib_ip_route_prefix_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router Id

← ***ipv4Prefix*** Virtual Router Id

← ***gateStr*** Gateway address

← ***distance*** Distance

← ***tag*** Tag

← ***desc*** Description

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.8 int smi_rib_ip_route_prefix_unset_sdkapi (struct smiclient_globals ∗ *azg*, int *vr_id*, char ∗ *ipv4_prefix*, char ∗ *gate_str*, int *distance*, u_int32_t *tag*, char ∗ *desc*)

Function to clear configured ipv4 static route. smi_rib_ip_route_prefix_unset_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router Id

← ***ipv4Prefix*** Ipv4 Address

← ***gateStr*** Gateway address

← ***distance*** Distance

← ***tag*** Tag

← ***desc*** Description

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.9 int smi_rib_ip_route_unset_sdkapi (struct smiclient_globals ∗ *azg*, int *vrId*, char ∗ *vrfName*, struct prefix_ipv4 ∗ *ipv4*)

Function to clear configured ipv4 static route. smi_rib_ip_route_unset_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router Id

← ***ipv4*** Ipv4 Address

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.10 int smi_rib_ipv4_route_set_sdkapi (struct smiclient_globals ∗ *azg*, int *vrId*, char ∗ *vrfName*, struct prefix_ipv4 ∗ *ipv4Prefix*, struct pal_in4_addr ∗ *ipv4GateAddr*, char ∗ *ifname*, int *distance*, int *metric*, int *snmpRouteType*, u_int32_t *tag*, char ∗ *desc*)

Establish ipv4 static routes into VRF. smi_rib_ipv4_route_set_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router Id

← ***vrfName*** VRF name

← ***ipv4Prefix*** Ipv4 Address

← ***ipv4GateAddr*** Gateway address

← ***ifName*** Interface name

← ***distance*** Distance

← ***metric*** Metric

← ***tag*** Tag

← ***desc*** Description

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.11 int smi_rib_ipv4_route_stale_clear (struct smiclient_globals ∗ *azg*, u_int32_t *vrId*)

Function to clear IPv4 stale kernel routes from NSM RIB and FIB. smi_rib_ipv4_route_stale_clear

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router Id

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.12 int smi_rib_ipv4_route_unset_sdkapi (struct smiclient_globals ∗ *azg*, int *vrId*, char ∗ *vrfName*, struct prefix_ipv4 ∗ *ipv4Prefix*, struct pal_in4_addr ∗ *ipv4GateAddr*, char ∗ *ifname*, int *distance*, u_int32_t *tag*, char ∗ *desc*)

clear ipv4 static routes into VRF smi_rib_ipv4_route_unset_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router Id

← ***vrfId*** VRF name

← ***ipv4Prefix*** Ipv4 Address

← ***ipv4GateAddr*** Gateway address

← ***ifName*** Interface name

← ***distance*** Distance

← ***metric*** Metric

← ***tag*** Tag

← ***desc*** Description

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.13 int smi_rib_ipv4_route_vrf_ifname_set_sdkapi (struct smiclient_globals ∗ *azg*, int *vrId*, char ∗ *vrfName*, struct prefix_ipv4 ∗ *ipv4Prefix*, struct pal_in4_addr ∗ *ipv4GateAddr*, char ∗ *ifName*, int *distance*, int *metric*, u_int32_t *tag*, char ∗ *desc*)

Establish ipv4 static routes into VRF interfacename. smi_rib_ipv4_route_vrf_ifname_set_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router Id

← ***vrfName*** VRF name

← ***ipv4Prefix*** Ipv4 Address

← ***ipv4GateAddr*** Gateway address

← ***ifName*** Interface name

← ***distance*** Distance

← ***metric*** Metric

← ***tag*** Tag

← ***desc*** Description

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.14 int smi_rib_ipv4_route_vrf_ifname_unset_sdkapi (struct smiclient_globals ∗ *azg*, int *vrId*, char ∗ *vrfName*, struct prefix_ipv4 ∗ *ipv4Prefix*, struct pal_in4_addr *ipv4GateAddr*, char ∗ *ifName*, int *distance*, u_int32_t *tag*, char ∗ *desc*)

clear ipv4 static routes into VRF interface name smi_rib_ipv4_route_vrf_ifname_unset_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router Id

← ***vrfName*** VRF name

← ***ipv4Prefix*** Ipv4 Address

← ***ipv4GateAddr*** Gateway address (Pass null in case of no gateway address)

← ***ifName*** Interface name

← ***distance*** Distance

← ***metric*** Metric

← ***tag*** Tag

← ***desc*** Description

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.15 int smi_rib_ipv6_route_stale_clear (struct smiclient_globals ∗ *azg*, u_int32_t *vrId*)

Function to clear IPv6 stale kernel routes from NSM RIB and FIB. smi_rib_ipv6_route_stale_clear

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router Id

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.16 int smi_rib_multipath_num_func_sdkapi (struct smiclient_globals ∗ *azg*, int *set*, int *mutipathNum*)

Set multipath numbers installed to FIB. smi_rib_multipath_num_func_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***set*** Flag to determine set or unset. Pass O to set multipath number to default

← ***mutipathNum*** Number of multipath to be set

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.17 int smi_rib_no_debug (struct smiclient_globals ∗ *azg*, int *vrId*, int *debug*)

Use this function to disable specific debugging. smi_rib_no_debug

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router Id

← ***debug*** Pass debug flag as following:

SMI_RIB_DBG_ALL - Debug all RIB information

SMI_RIB_DBG_EVENTS - Debug RIB events

SMI_RIB_DBG_PACKET - Debug RIB and NSM communications

SMI_RIB_DBG_PACKET_SEND - Debug sent packets

SMI_RIB_DBG_PACKET_RECV - Debug received packets

SMI_RIB_DBG_PACKET_DETAIL - Display detailed information for the sent and received packet

SMI_RIB_DBG_PACKET_SEND_DETAIL - Display detailed information for the sent packet

SMI_RIB_DBG_PACKET_RECV_DETAIL - Display detailed information for the received packet

Returns:

0 on success, otherwise one of the following error codes RIB_API_SET_ERR_-VR_NOT_EXIST

2.1.2.18 int smi_rib_set_maximum_fib_routes_sdkapi (struct smiclient_globals ∗ *azg*, int *vrId*, int *num*)

Set maximum fib routes number. Allowed number of fib routes excluding Kernel, Connect and Static. smi_rib_set_maximum_fib_routes_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router ID

← ***num*** Fib routes number to be set. Range <1-4294967294>

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.19 int smi_rib_set_maximum_static_routes_sdkapi (struct smiclient_globals ∗ *azg*, int *vrId*, int *num*)

Set maximum static routes number. smi_rib_set_maximum_static_routes_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router ID

← ***num*** Number of the static route which can be configured <1-4294967294>

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.20 int smi_rib_show_ipv4_route_details_sdkapi (struct smiclient_globals ∗ *azg*, int *startIndex*, int *endIndex*, char ∗ *vrfName*, int *vrId*, char ∗ *ipAddr*, struct list ∗ *ribList*, int(∗)(struct list ∗ribList) *callback*)

Use this function to get ipv4 routing table for the specified ip address. smi_rib_show_-ipv4_route_details_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***startIndex*** start index

← ***endIndex*** end index

← ***vrfName*** VRF Name. Pass NULL for default VRF.

← ***vrId*** Virtual Router Id

← ***ipAddr*** ipv4 address

→ ***ribList*** Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.

→ ***callback*** Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes SMI_ERROR

2.1.2.21 int smi_rib_show_ipv4_route_interface (struct smiclient_globals ∗ *azg*, int *startIndex*, int *endIndex*, char ∗ *vrfName*, int *vrId*, char ∗ *ifName*, struct list ∗ *ribList*, int(∗)(struct list ∗ribList) *callback*)

Use this function to get all routes from the routing table for specified interface name. smi_rib_show_ipv4_route_interface

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***startIndex*** start index

← ***endIndex*** end index

← ***vrfName*** VRF Name. Pass NULL for default VRF.

← ***vrId*** Virtual Router Id

← ***ifName*** Interface Name

→ ***ribList*** Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.

→ ***callback*** Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.22 int smi_rib_show_ipv4_route_nhaddr (struct smiclient_globals ∗ *azg*, int *startIndex*, int *endIndex*, char ∗ *vrfName*, int *vrId*, char ∗ *ipAddr*, struct list ∗ *ribList*, int(∗)(struct list ∗ribList) *callback*)

Use this function to get all ipv4 routes from the routing table for specified next hop address. smi_rib_show_ipv4_route_nhaddr

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***startIndex*** start index

← ***endIndex*** end index

← ***vrfName*** VRF Name. Pass NULL for default VRF.

← ***vrId*** Virtual Router Id

← ***ipAddr*** Next hop in the IPV4 routing table to display

→ ***ribList*** Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.

→ ***callback*** Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.23 int smi_rib_show_ipv6_route_details_sdkapi (struct smiclient_globals ∗ *azg*, int *startIndex*, int *endIndex*, char ∗ *vrfName*, int *vrId*, char ∗ *ipAddr*, struct list ∗ *ribList*, int(∗)(struct list ∗ribList) *callback*)

Use this function to get ipv6 routing table for the specified ip address. smi_rib_show_ipv6_route_details_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***startIndex*** start index

← ***endIndex*** end index

← ***vrfName*** VRF Name. Pass NULL for default VRF.

← ***vrId*** Virtual Router Id

← ***ipAddr*** ipv6 address

→ ***ribList*** Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.

→ ***callback*** Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.24 int smi_rib_show_ipv6_route_interface (struct smiclient_globals ∗ *azg*, int *startIndex*, int *endIndex*, char ∗ *vrfName*, int *vrId*, char ∗ *ifName*, struct list ∗ *ribList*, int(∗)(struct list ∗ribList) *callback*)

Use this function to get all ipv6 routes from the routing table for specified interface name. smi_rib_show_ipv6_route_interface

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***startIndex*** start index

← ***endIndex*** end index

← ***vrfName*** VRF Name. Pass NULL for deafult VRF

← ***vrId*** Virtual Router Id

← ***ifname*** Interface Name

→ ***ribList*** Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.

→ ***callback*** Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.25 int smi_rib_show_ipv6_route_nhaddr (struct smiclient_globals ∗ *azg*, int *startIndex*, int *endIndex*, char ∗ *vrfName*, int *vrId*, char ∗ *ipAddr*, struct list ∗ *ribList*, int(∗)(struct list ∗ribList) *callback*)

Use this function to get all ipv6 routes from the routing table for specified next hop address. smi_rib_show_ipv6_route_nhaddr

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***startIndex*** start index

← ***endIndex*** end index

← ***vrfName*** VRF Name. Pass NULL for default VRF.

← ***vrId*** Virtual Router Id

← ***ipAddr*** Next hop in the IPV6 routing table to display

→ ***ribList*** Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.

→ ***callback*** Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.26 int smi_rib_show_route_ipv4_sdkapi (struct smiclient_globals ∗ *azg*, int *startIndex*, int *endIndex*, int *vrId*, char ∗ *vrfName*, int *database*, u_char *type*, struct list ∗ *ribList*, int(∗)(struct list ∗ribList) *callback*)

Use this function to get all ipv4 routes from the routing table. smi_rib_show_route_ipv4_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***startIndex*** start index

← ***endIndex*** end index

← ***vrId*** Virtual Router Id

← ***vrfName*** VRF Name. Pass NULL to get all VRF details in a list.

← ***database*** To Display IPv6 routing table database information pass '1' , else '0'

← ***type*** Type of the route to be fetched <1-9> 1-IPI_ROUTE_KERNEL, 2-IPI_ROUTE_CONNECT
3-IPI_ROUTE_STATIC, 4-IPI_ROUTE_RIP
5-IPI_ROUTE_RIPNG, 6-IPI_ROUTE_OSPF
7-IPI_ROUTE_OSPF6, 8-IPI_ROUTE_BGP
9-IPI_ROUTE_ISIS

→ ***ribList*** Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.

→ ***callback*** Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.27 int smi_rib_show_route_ipv6_sdkapi (struct smiclient_globals ∗ *azg*, int *startIndex*, int *endIndex*, int *vrId*, char ∗ *vrfName*, int *database*, u_char *type*, struct list ∗ *ribList*, int(∗)(struct list ∗ribList) *callback*)

Function to establish IPV6 static routes into VRF. smi_ipv6_route_vrf_ifname_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router Id

← ***vrfName*** VRF Name

← ***ipv6Prefix*** Ipv6 Address

← ***gateStr*** Gateway address

← ***distance*** Distance

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

smi_rib_show_route_ipv6_sdkapi

Use this function to get all ipv6 routes from the routing table.

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***startIndex*** start index

← ***endIndex*** end index

← ***vrId*** Virtual Router Id

← ***vrfName*** VRF Name. Pass NULL to get all VRF details in a list.

← ***database*** To Display IPv6 routing table database information pass ’1’ , else pass ’0’

← ***type*** Type of the route to be fetched <1-9> 1-IPI_ROUTE_KERNEL, 2-IPI_ROUTE_CONNECT
3-IPI_ROUTE_STATIC, 4-IPI_ROUTE_RIP
5-IPI_ROUTE_RIPNG, 6-IPI_ROUTE_OSPF
7-IPI_ROUTE_OSPF6, 8-IPI_ROUTE_BGP
9-IPI_ROUTE_ISIS

→ ***ribList*** Link list of structure smi_rib. smi_rib structure holds details of single route. List should be intialized by caller.

→ ***callback*** Callback function which take list as input parameter, here the list will be containing the nodes of type structure smi_rib. Pass NULL in case of no callback function required.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.28 int smi_rib_show_route_summary_ipv4 (struct smiclient_globals ∗ *azg*, char ∗ *vrfName*, int *vrId*, struct smi_route_summ ∗ *rtSumm*)

Function to get summary of all ipv4 routes. smi_rib_show_route_summary_ipv4

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrfName*** VRF Name. Pass NULL for default VRF.

← ***vrId*** Virtual Router Id

→ ***rtSumm*** Output parameter to hold the route summary smi_route_summ.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.29 int smi_rib_show_route_summary_ipv6 (struct smiclient_globals ∗ *azg*, char ∗ *vrfName*, int *vrId*, struct smi_route_summ ∗ *rtSumm*)

Function to get summary of all ipv6 routes. smi_rib_show_route_summary_ipv6

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrfName*** VRF Name. Pass NULL for default VRF.

← ***vrId*** Virtual Router Id

→ ***rtSumm*** Output parameter to hold the route summary smi_route_summ.

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.30 int smi_rib_unset_maximum_fib_routes_sdkapi (struct smiclient_globals ∗ *azg*, int *vrId*)

Set maximum fib routes number to the default value 4294967294. smi_rib_unset_-maximum_fib_routes_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router ID

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

2.1.2.31 int smi_rib_unset_maximum_static_routes_sdkapi (struct smiclient_globals ∗ *azg*, int *vrId*)

Set maximum static routes number to the default value 4294967294UL. smi_rib_-unset_maximum_static_routes_sdkapi

Parameters:

← ***azg*** Pointer to the SMI client global structure

← ***vrId*** Virtual Router ID

Returns:

SMI_SUCCESS on success, otherwise one of the following error codes
SMI_ERROR

Index