ZebOS-XP ARP SMI Reference

IP Infusion Inc.

Generated by Doxygen 1.6.1

Wed Dec 16 12:33:20 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_arp.h File Reference

Provides API for managing ARP. `#include "smi_client.h"`

`#include "smi_arp_msg.h"`

Functions

- int smi_nsm_api_arp_entry_add_sdkapi (struct smiclient_globals *azg, u_int32_t vr_id, struct pal_in4_addr *addr, char *mac_addr, u_int8_t is_proxy_arp)

  This function creates a static proxy ARP entry. This function implements the arp A.B.C.D MAC command.

- int smi_get_all_arp_entry (struct smiclient_globals *azg, int vr_id, int start_index, int end_index, struct list *arplist, int(*funPointer)(struct list *arplist))

  This function get all ipv4 arp entries.

- int smi_nsm_api_arp_entry_del_sdkapi (struct smiclient_globals *azg, u_int32_t vr_id, struct pal_in4_addr *addr, unsigned char *mac_addr, char *ifname)

  This function deletes an ARP entry. This function implements the no arp A.B.C.D MAC commandfunction implements the arp A.B.C.D MAC command.

- int smi_nsm_api_arp_del_all_sdkapi (struct smiclient_globals *azg, u_int32_t vr_id, u_char clr_flag)

  This function clear ARP Cache.

- int smi_nsm_if_arp_ageing_timeout_set (struct smiclient_globals *azg, u_int32_t vr_id, char *ifname, int arp_ageing_timeout)

  This function will set ageing timout value for ARP.

- int smi_nsm_if_arp_ageing_timeout_get (struct smiclient_globals ∗azg, u_int32_t vr_id, char ∗ifname, int ∗arp_ageing_timeout)

  This function will get ageing timout value for ARP.

- int smi_nsm_if_arp_ageing_timeout_unset (struct smiclient_globals ∗azg, u_int32_t vr_id, char ∗ifname)

  This function will unset ageing timout value for ARP.

- int smi_ipv6_nbr_get_all (struct smiclient_globals ∗azg, int vr_id, int start_index, int end_index, struct list ∗arplist, int(∗funPointer)(struct list ∗arplist))

  This function is to add an IPv6 neighbor entry.

- int smi_nsm_api_ipv6_nbr_del_all_sdkapi (struct smiclient_globals ∗azg, u_int32_t vr_id, u_char clr_flag)

  This function is to remove an IPv6 neighbor entry.

2.1.1 Detailed Description

Provides API for managing ARP.

2.1.2 Function Documentation

2.1.2.1 int smi_get_all_arp_entry (struct smiclient_globals ∗ *azg*, int *vr_id*, int *start_index*, int *end_index*, struct list ∗ *arplist*, int(∗)(struct list ∗arplist) *funPointer*)

This function get all ipv4 arp entries. smi_get_all_arp_entry

Parameters:

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

← ***vr_id*** Virtual Router Id

← ***start_index*** start index

← ***end_index*** end index

→ ***arplist*** Link list of structure smi_arp_entry. smi_arp_entry structure holds details of a specific arp entry. 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_arp_entry. Pass NULL in case of no callback function required.

Returns:

NSM_SUCCESS when the function succeeds, otherwise one of the following error codes
NSM_API_SET_ARP_GENERAL_ERR

2.1.2.2 int smi_ipv6_nbr_get_all (struct smiclient_globals ∗ *azg*, int *vr_id*, int *start_index*, int *end_index*, struct list ∗ *arplist*, int(∗)(struct list ∗arplist) *funPointer*)

This function is to add an IPv6 neighbor entry. smi_nsm_api_ipv6_nbr_add_sdkapi

Parameters:

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

← ***vr_id*** Virtual Router Id

← ***ipv6_prefix*** Specify the neighbor's IPv6 address

← ***ifname*** Interface name

← ***mac_addr*** MAC address

Returns:

NSM_SUCCESS when the function succeeds, otherwise one of the following error codes
NSM_API_SET_ERR_INVALID_VALUE
NSM_API_SET_ARP_GENERAL_ERR
NSM_API_SET_ERROR
NSM_API_SET_ERR_HAL_FAILURE

smi_ipv6_nbr_get_all This function get all ipv6 neighbor entry

Parameters:

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

← ***vr_id*** Virtual Router Id

← ***start_index*** start index

← ***end_index*** end index

→ ***arplist*** Link list of structure smi_arp_entry. smi_arp_entry structure holds details of a specific arp entry. 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_arp_entry. Pass NULL in case of no callback function required.

Returns:

NSM_SUCCESS when the function succeeds, otherwise one of the following error codes
NSM_API_SET_ARP_GENERAL_ERR

2.1.2.3 int smi_nsm_api_arp_del_all_sdkapi (struct smiclient_globals ∗ *azg*, u_int32_t *vr_id*, u_char *clr_flag*)

This function clear ARP Cache. smi_nsm_api_arp_del_all_sdkapi

Parameters:

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

← ***vr_id*** Virtual router ID

← ***clr_flag*** Clear flag (static | dynamic)

Returns:

NSM_SUCCESS when the function succeeds.

2.1.2.4 int smi_nsm_api_arp_entry_add_sdkapi (struct smiclient_globals * *azg*, u_int32_t *vr_id*, struct pal_in4_addr * *addr*, char * *mac_addr*, u_int8_t *is_proxy_arp*)

This function creates a static proxy ARP entry. This function implements the arp A.B.C.D MAC command. smi_nsm_api_arp_entry_add_sdkapi

Parameters:

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

← ***vr_id*** Virtual router ID

← ***addr*** IP address

← ***mac_addr*** Hardware address

← ***is_proxy_arp*** enable (1) | disable (0)

Returns:

NSM_SUCCESS when the function succeeds, otherwise one of the following error codes
NSM_API_SET_ERR_INVALID_VALUE
NSM_API_SET_ARP_GENERAL_ERR
NSM_API_SET_ERR_MAX_STATIC_ARP_LIMIT_EXCEEDED
NSM_API_SET_ERR_MAX_VR_STATIC_ARP_LIMIT_EXCEEDED

2.1.2.5 int smi_nsm_api_arp_entry_del_sdkapi (struct smiclient_globals * *azg*, u_int32_t *vr_id*, struct pal_in4_addr * *addr*, unsigned char * *mac_addr*, char * *ifname*)

This function deletes an ARP entry. This function implements the no arp A.B.C.D MAC commandfunction implements the arp A.B.C.D MAC command. smi_nsm_-api_arp_entry_del_sdkapi

Parameters:

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

← ***vr_id*** Virtual router ID

← ***addr*** IP address

← ***mac_addr*** Hardware address

← ***ifname*** Interface name

Returns:

NSM_SUCCESS when the function succeeds, otherwise one of the following error codes
NSM_API_SET_ERR_INVALID_VALUE
NSM_API_SET_ARP_GENERAL_ERR

2.1.2.6 int smi_nsm_api_ipv6_nbr_del_all_sdkapi (struct smiclient_globals * *azg*, u_int32_t *vr_id*, u_char *clr_flag*)

This function is to remove an IPv6 neighbor entry. smi_nsm_api_ipv6_nbr_del_sdkapi

Parameters:

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

← ***vr_id*** Virtual Router Id

← ***ipv6_prefix*** Specify the neighbor's IPv6 address

← ***ifname*** Interface name

← ***clr_flag*** NSM_API_ARP_FLAG_STATIC(1) or NSM_API_ARP_FLAG_-DYNAMIC (2)

Returns:

NSM_SUCCESS when the function succeeds, otherwise one of the following error codes
NSM_API_SET_ERR_INVALID_VALUE
NSM_API_SET_ARP_GENERAL_ERR
NSM_API_SET_ERROR
NSM_API_SET_ERR_HAL_FAILURE

smi_nsm_api_ipv6_nbr_del_all_sdkapi This function is to remove all IPv6 neighbor entry.a

Parameters:

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

← ***vr_id*** Virtual Router Id

← ***clr_flag*** NSM_API_ARP_FLAG_STATIC(1) or NSM_API_ARP_FLAG_-DYNAMIC (2)

Returns:

NSM_SUCCESS when the function succeeds

2.1.2.7 int smi_nsm_if_arp_ageing_timeout_get (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *ifname*, int * *arp_ageing_timeout*)

This function will get ageing timout value for ARP. smi_nsm_if_arp_ageing_timeout_-get

Parameters:

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

← ***vr_id*** Virtual router ID

← ***ifname*** Interface name

→ ***arp_ageing_timeout*** Ageing timeout <1-3000>

Returns:

NSM_API_SET_SUCCESS when the function succeeds, otherwise one of the following error codes
NSM_API_SET_ERR_VR_NOT_EXIST
NSM_API_SET_ERR_IF_NOT_EXIST

2.1.2.8 int smi_nsm_if_arp_ageing_timeout_set (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *ifname*, int *arp_ageing_timeout*)

This function will set ageing timout value for ARP. smi_nsm_if_arp_ageing_timeout_-set

Parameters:

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

← ***vr_id*** Virtual router ID

← ***ifname*** Interface name

← ***arp_ageing_timeout*** Ageing timeout <1-3000>

Returns:

NSM_API_SET_SUCCESS when the function succeeds, otherwise one of the following error codes
NSM_API_SET_ERR_VR_NOT_EXIST
NSM_API_SET_ERR_IF_NOT_EXIST
NSM_API_SET_ERR_ARP_AGEING_TIMEOUT

2.1.2.9 int smi_nsm_if_arp_ageing_timeout_unset (struct smiclient_globals * *azg*, u_int32_t *vr_id*, char * *ifname*)

This function will unset ageing timout value for ARP. smi_nsm_if_arp_ageing_-timeout_unset

Parameters:

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

← ***vr_id*** Virtual router ID

← ***ifname*** Interface name

Returns:

NSM_API_SET_SUCCESS when the function succeeds, otherwise one of the following error codes
NSM_API_SET_ERR_VR_NOT_EXIST
NSM_API_SET_ERR_IF_NOT_EXIST
NSM_API_SET_ERR_ARP_AGEING_TIMEOUT

Index