ZebOS-XP Private VLAN SMI Reference

IP Infusion Inc.

Generated by Doxygen 1.6.1

Wed Dec 16 12:33:26 2015

Contents

Chapter 1

Data Structure Index

1.1 Data Structures

Here are the data structures with brief descriptions:

Chapter 2

File Index

2.1 File List

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

Chapter 3

Data Structure Documentation

3.1 pSecondaryVlan Struct Reference

Data Fields

- u_int16_t **secondaryVid**
- enum secondaryVlanType **secondaryVlanType**

The documentation for this struct was generated from the following file:

- smi_pvlan_msg.h

3.2 pvlan_msg_ Struct Reference

Data Fields

- smi_cindex_t **cindex_0**
- u_int32_t **vr_id**
- char **br_id** [255]
- u_int16_t **vlan_id**
- u_int8_t **pvlan_type**
- u_int16_t **pvid**
- char **ifname** [255]
- char **bridge_name** [255]
- u_int8_t **mode**
- struct smi_vlan_bmp **pvid_bmp**
- int **primaryVid**
- struct pvlanPrimList **primData**
- struct pvlanSeconList **seconData**

The documentation for this struct was generated from the following file:

- smi_pvlan_msg.h

3.3 pvlanPrimList Struct Reference

Data Fields

- int **prim_more**
- int **prim_count**
- struct list * **pSeconVlanList**

The documentation for this struct was generated from the following file:

- smi_pvlan_msg.h

3.4 pvlanSeconList Struct Reference

Data Fields

- int **secon_more**
- int **secon_count**
- struct list * **pSeconVlanList**

The documentation for this struct was generated from the following file:

- smi_pvlan_msg.h

Chapter 4

File Documentation

4.1 smi_pvlan.h File Reference

A private VLAN contains switch ports that can not communicate with each other, but can access other networks. This file provides APIs for the configuration and management of private VLANs within ZebOS. `#include "smi_client.h"`

`#include "smi_pvlan_msg.h"`

Functions

- int **smi_pvlan_host_association_add_validate** (struct smiclient_globals *azg, u_int32_t vrId, char *ifName, u_int16_t pvlanId, u_int16_t svlanId)
- int **smi_pvlan_set_port_mode_validate** (struct smiclient_globals *azg, u_int32_t vrId, char *ifName, enum smi_pvlan_port_mode portMode)
- int **smi_pvlan_associate_clear_all_validate** (struct smiclient_globals *azg, u_int32_t vrId, char *bridgeId, u_int16_t vlanId)
- int **smi_pvlan_associate_remove_validate** (struct smiclient_globals *azg, u_int32_t vrId, char *bridgeId, u_int16_t pvlanId, u_int16_t svlanId)
- int **smi_pvlan_associate_add_validate** (struct smiclient_globals *azg, u_int32_t vrId, char *bridgeId, u_int16_t pvlanId, u_int16_t svlanId)
- int **smi_pvlan_delete_validate** (struct smiclient_globals *azg, u_int32_t vrId, char *bridgeId, u_int16_t vlanId, enum smi_pvlan_type pvlan_type)
- int **smi_pvlan_create_validate** (struct smiclient_globals *azg, u_int32_t vrId, char *bridgeId, u_int16_t vlanId, enum smi_pvlan_type pvlanType)
- int **smi_pvlan_switchport_mapping_add_validate** (struct smiclient_globals *azg, u_int32_t vrId, char *ifName, u_int16_t vlanId, struct smi_vlan_bmp *pvlanIdBmp)
- int **smi_pvlan_switchport_mapping_remove_validate** (struct smiclient_globals *azg, u_int32_t vrId, char *ifName, u_int16_t vlanId, struct smi_vlan_bmp *pvlanIdBmp)
- int smi_pvlan_create (struct smiclient_globals *azg, u_int32_t vrId, char *bridgeId, u_int16_t vlanId, enum smi_pvlan_type pvlanType)

Creates a private VLAN.

- int smi_pvlan_delete (struct smiclient_globals *azg, u_int32_t vrId, char *bridgeId, u_int16_t vlanId, enum smi_pvlan_type pvlanType)

  Deletes a private VLAN.

- int smi_pvlan_associate_add (struct smiclient_globals *azg, u_int32_t vrId, char *bridgeId, u_int16_t pvlanId, u_int16_t svlanId)

  Associates a secondary VLAN to a primary VLAN. Only one isolated VLAN can be associated to a primary VLAN. Multiple community VLAN can be associated to a primary VLAN.

- int smi_pvlan_associate_remove (struct smiclient_globals *azg, u_int32_t vrId, char *bridgeId, u_int16_t pvlanId, u_int16_t svlanId)

  Remove the associate between primary and secondary VLAN.

- int smi_pvlan_associate_clear_all (struct smiclient_globals *azg, u_int32_t vrId, char *bridgeId, u_int16_t vlanId)

  Remove association of all the secondary VLAN to primary VLAN.

- int smi_pvlan_set_port_mode (struct smiclient_globals *azg, u_int32_t vrId, char *ifName, enum smi_pvlan_port_mode portMode)

  Set a layer 2 port as host port or promiscous port.

- int smi_pvlan_get_port_mode (struct smiclient_globals *azg, u_int32_t vrId, char *ifName, enum smi_pvlan_port_mode *portMode)

  Get a layer 2 port as host port or promiscous port.

- int smi_pvlan_clear_port_mode_validate (struct smiclient_globals *azg, u_int32_t vrId, char *ifName, enum smi_pvlan_port_mode portMode)

  Remove host/promiscous portMode configuration from a port.

- int **smi_pvlan_clear_port_mode** (struct smiclient_globals *azg, u_int32_t vrId, char *ifName, enum smi_pvlan_port_mode portMode)
- int smi_pvlan_host_association_add (struct smiclient_globals *azg, u_int32_t vrId, char *ifName, u_int16_t pvlanId, u_int16_t svlanId)

  Associate a primary VLAN and a secondary VLAN to a host port.

- int smi_pvlan_host_association_remove_validate (struct smiclient_globals *azg, u_int32_t vrId, char *ifName)

  Remove primary and secondary VLAN association to a host port.

- int **smi_pvlan_host_association_remove** (struct smiclient_globals *azg, u_int32_t vrId, char *ifName)
- int **smi_pvlan_host_association_get** (struct smiclient_globals *azg, u_int32_t vrId, char *ifName, u_int16_t pvlanId, u_int16_t *svlanId)

- int smi_pvlan_switchport_mapping_add (struct smiclient_globals *azg, u_int32_t vrId, char *ifName, u_int16_t vlanId, struct smi_vlan_bmp *pvlanIdBmp)

  Asscoiate a primary VLAN and a set of secondary VLANs to a promiscous port.

- int smi_pvlan_switchport_mapping_remove (struct smiclient_globals *azg, u_int32_t vrId, char *ifName, u_int16_t vlanId, struct smi_vlan_bmp *pvlanIdBmp)

  Remove the association of a set of secondary VLANs to a primary VLAN on a promiscous port.

- int smi_pvlan_switchport_mapping_remove_all (struct smiclient_globals *azg, u_int32_t vrId, char *ifName)

  Remove all the association of secondary VLANs to primary VLAN for a promiscous port.

- int smi_nsm_show_private_vlan (struct smiclient_globals *azg, u_int32_t vrId, char *bridgeId, int *pvlanId, struct list *primList, int(*funPointer1)(struct list *primList), struct list *seconList, int(*funPointer2)(struct list *seconList))

  Shows all the private VLAN's information of given bridge.

- int smi_nsm_show_private_vlan_default_bridge (struct smiclient_globals *azg, u_int32_t vrId, int *pvlanId, struct list *primList, int(*funPointer1)(struct list *primList), struct list *seconList, int(*funPointer2)(struct list *seconList))

  Shows all the private VLAN's information of default bridge.

- int **smi_client_create_n_send_pvlan_msg** (struct smi_client_handler *async, int vrid, pvlan_msg *msg, int optype)
- int **smi_client_read_sync_pvlan_msg** (struct smi_client_handler *ach, int msgtype, void *getmsg)

4.1.1 Detailed Description

A private VLAN contains switch ports that can not communicate with each other, but can access other networks. This file provides APIs for the configuration and management of private VLANs within ZebOS. nsm

4.1.2 Function Documentation

4.1.2.1 int smi_nsm_show_private_vlan (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *bridgeId*, int * *pvlanId*, struct list * *primList*, int(*)(struct list *primList) *funPointer1*, struct list * *seconList*, int(*)(struct list *seconList) *funPointer2*)

Shows all the private VLAN's information of given bridge. smi_nsm_show_private_-vlan

Parameters:

← ***azg*** Pointer to the SMI client global structure
← ***vrId*** Virtual Router ID <0-255>
← ***bridgeId*** Bridge name <1-32>
→ ***pvlanId*** Pimary Vlan Id
→ ***primList*** Pointer to linked list of pvlanId
← ***funPointer1*** Callback function pointer
→ ***seconList*** Pointer to linked list of structure pSecondaryVlan
← ***funPointer2*** Callback function pointer

Returns:

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

4.1.2.2 int smi_nsm_show_private_vlan_default_bridge (struct smiclient_globals * *azg*, u_int32_t *vrId*, int * *pvlanId*, struct list * *primList*, int(*)(struct list *primList) *funPointer1*, struct list * *seconList*, int(*)(struct list *seconList) *funPointer2*)

Shows all the private VLAN's information of default bridge. smi_nsm_show_private_-vlan_default_bridge

Parameters:

← ***azg*** Pointer to the SMI client global structure
← ***vrId*** Virtual Router ID <0-255>
← ***bridgeId*** Bridge name <1-32>
→ ***pvlanId*** Pimary Vlan Id
→ ***primList*** Pointer to linked list of pvlanId
← ***funPointer1*** Callback function pointer
→ ***seconList*** Pointer to linked list of structure pSecondaryVlan
← ***funPointer2*** Callback function pointer

Returns:

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

4.1.2.3 int smi_pvlan_associate_add (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *bridgeId*, u_int16_t *pvlanId*, u_int16_t *svlanId*)

Associates a secondary VLAN to a primary VLAN. Only one isolated VLAN can be associated to a primary VLAN. Multiple community VLAN can be associated to a primary VLAN. smi_pvlan_associate_add

Parameters:

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

← ***vrId*** Virtual Router ID

← ***bridgeId*** Bridge Id <1-32>

← ***vlanId*** VLAN Id <2-4094>

← ***pvlanId*** Private VLAN identifier <2-4094>

Returns:

0 on success, otherwise one of the following error codes
NSM_VLAN_ERR_BRIDGE_NOT_FOUND
NSM_VLAN_ERR_BRIDGE_NOT_VLAN_AWARE
NSM_NO_VLAN_CONFIGURED
NSM_PVLAN_ERR_PROVIDER_BRIDGE
NSM_PVLAN_ERR_PRIMARY_SECOND_SAME
NSM_VLAN_ERR_VLAN_NOT_FOUND
NSM_PVLAN_ERR_NOT_CONFIGURED
NSM_PVLAN_ERR_NOT_PRIMARY_VLAN
NSM_PVLAN_ERR_NOT_SECONDARY_VLAN
NSM_PVLAN_ERR_ISOLATED_VLAN_EXISTS
NSM_PVLAN_ERR_ASSOCIATED_TO_PRIMARY

4.1.2.4 int smi_pvlan_associate_clear_all (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *bridgeId*, u_int16_t *vlanId*)

Remove association of all the secondary VLAN to primary VLAN. smi_pvlan_-associate_clear_all

Parameters:

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

← ***vrId*** Virtual Router Id

← ***bridgeId*** Bridge Id <1-32>

← ***vlanId*** VLAN Id <2-4094>

Returns:

0 on success, otherwise one of the following error codes
NSM_VLAN_ERR_BRIDGE_NOT_FOUND
NSM_VLAN_ERR_BRIDGE_NOT_VLAN_AWARE
NSM_NO_VLAN_CONFIGURED
NSM_PVLAN_ERR_PROVIDER_BRIDGE
NSM_VLAN_ERR_VLAN_NOT_FOUND
NSM_PVLAN_NOT_CONFIGURED
NSM_PVLAN_ERR_NOT_PRIMARY_VLAN

4.1.2.5 int smi_pvlan_associate_remove (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *bridgeId*, u_int16_t *pvlanId*, u_int16_t *svlanId*)

Remove the associate between primary and secondary VLAN. smi_pvlan_associate_-remove

Parameters:

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

← ***vrId*** Virtual Router Id

← ***bridgeId*** Bridge Id <1-32>

← ***vlanId*** VLAN Id <2-4094>

← ***pvlanId*** Private VLAN Id <2-4094>

0 0n success, otherwise one of the following error codes

NSM_VLAN_ERR_BRIDGE_NOT_FOUND

NSM_VLAN_ERR_BRIDGE_NOT_VLAN_AWARE

NSM_NO_VLAN_CONFIGURED

NSM_PVLAN_ERR_PROVIDER_BRIDGE

NSM_VLAN_ERR_VLAN_NOT_FOUND

NSM_PVLAN_ERR_NOT_CONFIGURED

NSM_PVLAN_ERR_NOT_PRIMARY_VLAN

NSM_PVLAN_ERR_NOT_SECONDARY_VLAN

NSM_VLAN_ERR_VLAN_NOT_FOUND

4.1.2.6 int smi_pvlan_clear_port_mode_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *ifName*, enum smi_pvlan_port_mode *portMode*)

Remove host/promiscous portMode configuration from a port. smi_pvlan_clear_port_-mode

Parameters:

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

← ***vrId*** Virtual Router Id

← ***ifName*** Interface Name from which host/promiscous portMode configuration needs to be removed

← ***portMode*** Port mode to be removed host/promiscous

0 on success, otherwise one of the following error codes

NSM_VLAN_ERR_IFP_NOT_BOUND

NSM_VLAN_ERR_BRIDGE_NOT_FOUND

NSM_PVLAN_ERR_PROVIDER_BRIDGE

NSM_VLAN_ERR_INVALID_MODE

NSM_PVLAN_ERR_INVALID_MODE

4.1.2.7 int smi_pvlan_create (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *bridgeId*, u_int16_t *vlanId*, enum smi_pvlan_type *pvlanType*)

Creates a private VLAN. smi_pvlan_create

Parameters:

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

← ***vrId*** Virtual Router Id

← ***bridgeId*** Bridge Id <1-32>. If only default bridge is supported this value should be passed as 0.

← ***vlanId*** VLAN ID

← ***pvlanType*** It can take one of the following values
SMI_PVLAN_NONE = 0,
SMI_PVLAN_COMMUNITY = 1,
SMI_PVLAN_ISOLATED = 2,
SMI_PVLAN_PRIMARY = 3

Returns:

0 if success, otherwise one of the following error codes NSM_VLAN_ERR_-BRIDGE_NOT_FOUND
NSM_VLAN_ERR_BRIDGE_NOT_VLAN_AWARE
NSM_NO_VLAN_CONFIGURED
NSM_PVLAN_ERR_PROVIDER_BRIDGE
NSM_VLAN_ERR_VLAN_NOT_FOUND

4.1.2.8 int smi_pvlan_delete (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *bridgeId*, u_int16_t *vlanId*, enum smi_pvlan_type *pvlanType*)

Deletes a private VLAN. smi_pvlan_delete

Parameters:

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

← ***vrId*** Virtual Router Id

← ***bridgeId*** Bridge Name <1-32>

← ***vlanId*** VLAN Id

← ***pvlanType*** It can take one of the following values
SMI_PVLAN_NONE = 0,

SMI_PVLAN_COMMUNITY = 1,
SMI_PVLAN_ISOLATED = 2,
SMI_PVLAN_PRIMARY = 3

Returns:

0 on success, otherwise one of the following error codes NSM_VLAN_ERR_-BRIDGE_NOT_FOUND
NSM_VLAN_ERR_BRIDGE_NOT_VLAN_AWARE
NSM_NO_VLAN_CONFIGURED
NSM_PVLAN_ERR_PROVIDER_BRIDGE
NSM_VLAN_ERR_VLAN_NOT_FOUND
NSM_PVLAN_ERR_NOT_ISOLATED_VLAN
NSM_PVLAN_ERR_NOT_PRIMARY_VLAN
NSM_PVLAN_ERR_NOT_COMMUNTITY_VLAN
NSM_PVLAN_ERR_NOT_CONFIGURED

4.1.2.9 int smi_pvlan_get_port_mode (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *ifName*, enum smi_pvlan_port_mode * *portMode*)

Get a layer 2 port as host port or promiscous port. smi_pvlan_get_port_mode

Parameters:

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

← ***vrId*** Virtual Router Id

← ***ifName*** Interface name which needs to be set in host/promiscous portMode

→ ***portMode*** Port mode, can be one of the following NSM_PVLAN_PORT_-Mode_HOST = 1
NSM+PVLAN_PORT_Mode_PROMISCOUS = 2

Returns:

0 on success, otherwise one of the following error codes
NSM_VLAN_ERR_IFP_NOT_BOUND
NSM_VLAN_ERR_BRIDGE_NOT_FOUND
NSM_VLAN_ERR_BRIDGE_NOT_VLAN_AWARE
NSM_PVLAN_ERR_PROVIDER_BRIDGE
NSM_VLAN_ERR_INVALID_MODE
NSM_PVLAN_ERR_INVALID_MODE

4.1.2.10 int smi_pvlan_host_association_add (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *ifName*, u_int16_t *pvlanId*, u_int16_t *svlanId*)

Associate a primary VLAN and a secondary VLAN to a host port. smi_pvlan_host_-association_add

param[in] vrId Virtual Router Id

param[in] ifName Interface Name to which a primary and secondary VLANs needs to be associated

param[in] vlanId VLAN Id of the primary VLAN <2-4094>

param[in] pvlanId VLAN identifier of the secondary VLAN <2-4094>

Parameters:

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

Returns:

0 on success, otherwise one of the following error codes
NSM_VLAN_ERR_IFP_NOT_BOUND
NSM_VLAN_ERR_IFP_NOT_BOUND
NSM_PVLAN_ERR_PROVIDER_BRIDGE
NSM_VLAN_ERR_BRIDGE_NOT_VLAN_AWARE
NSM_VLAN_ERR_INVALID_MODE
NSM_PVLAN_ERR_NOT_HOST_PORT
NSM_NO_VLAN_CONFIGURED
NSM_VLAN_ERR_VLAN_NOT_FOUND
NSM_PVLAN_ERR_NOT_CONFIGURED
NSM_PVLAN_ERR_NOT_PRIMARY_VLAN
NSM_PVLAN_ERR_INVALID_MODE
NSM_PVLAN_ERR_SECOND_NOT_ASSOCIATED
NSM_VLAN_ERR_VLAN_NOT_FOUND

4.1.2.11 int smi_pvlan_host_association_remove_validate (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *ifName*)

Remove primary and secondary VLAN association to a host port. smi_pvlan_host_-association_remove

Parameters:

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

← ***vrId*** Virtual Router Id

← ***ifName*** Interface name from which primary and secondary VLAN association needs to be removed

Returns:

0 on success, otherwise one of the following error code
NSM_VLAN_ERR_IFP_NOT_BOUND
NSM_PVLAN_ERR_PROVIDER_BRIDGE
NSM_PVLAN_ERR_INVALID_MODE
NSM_VLAN_ERR_VLAN_NOT_FOUND
NSM_PVLAN_ERR_NOT_SECONDARY_VLAN
NSM_VLAN_ERR_VLAN_NOT_FOUND

4.1.2.12 int smi_pvlan_set_port_mode (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *ifName*, enum smi_pvlan_port_mode *portMode*)

Set a layer 2 port as host port or promiscous port. smi_pvlan_set_port_mode

Parameters:

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

← ***vrId*** Virtual Router Id

← ***ifName*** Interface name which needs to be set in host/promiscous portMode

← ***portMode*** Port mode, can be one of the following NSM_PVLAN_PORT_-Mode_HOST = 1
NSM+PVLAN_PORT_Mode_PROMISCOUS = 2

Returns:

0 on success, otherwise one of the following error codes
NSM_VLAN_ERR_IFP_NOT_BOUND
NSM_VLAN_ERR_BRIDGE_NOT_FOUND
NSM_VLAN_ERR_BRIDGE_NOT_VLAN_AWARE
NSM_PVLAN_ERR_PROVIDER_BRIDGE
NSM_VLAN_ERR_INVALID_MODE
NSM_PVLAN_ERR_INVALID_Mode

4.1.2.13 int smi_pvlan_switchport_mapping_add (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *ifName*, u_int16_t *vlanId*, struct smi_vlan_bmp * *pvlanIdBmp*)

Asscoiate a primary VLAN and a set of secondary VLANs to a promiscous port. smi_-pvlan_switchport_mapping_add

Parameters:

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

← ***vrId*** Virtual Router Id

← ***ifName*** Interface name to which primary and secondary VLANs needs to be associated

← ***vlanId*** primary VLAN id

← ***pvlanIdBmp*** A bitmap of secondary VLAN Ids

Returns:

0 on success, otherwise one of the following error codes
NSM_VLAN_ERR_IFP_NOT_BOUND
NSM_PVLAN_ERR_PROVIDER_BRIDGE
NSM_VLAN_ERR_BRIDGE_NOT_VLAN_AWARE
NSM_VLAN_ERR_INVALID_MODE

NSM_NO_VLAN_CONFIGURED
NSM_VLAN_ERR_VLAN_NOT_FOUND
NSM_PVLAN_ERR_NOT_CONFIGURED
NSM_PVLAN_ERR_NOT_PRIMARY_VLAN
NSM_PVLAN_ERR_INVALID_MODE
NSM_PVLAN_ERR_SECOND_NOT_ASSOCIATED

4.1.2.14 int smi_pvlan_switchport_mapping_remove (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *ifName*, u_int16_t *vlanId*, struct smi_vlan_bmp * *pvlanIdBmp*)

Remove the association of a set of secondary VLANs to a primary VLAN on a promiscous port. smi_pvlan_switchport_mapping_remove

Parameters:

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

← ***vrId*** Virtual Router Id

← ***ifName*** Interface name for which switchport mapping needs to be removed

← ***vlanId*** Primary VLAN id

← ***pvlanIdBmp*** A bitmap of secondary VLAN Ids

4.1.2.15 int smi_pvlan_switchport_mapping_remove_all (struct smiclient_globals * *azg*, u_int32_t *vrId*, char * *ifName*)

Remove all the association of secondary VLANs to primary VLAN for a promiscous port. smi_pvlan_switchport_mapping_remove_all

Parameters:

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

← ***vrId*** Virtual Router Id

← ***ifName*** Interface Name

Returns:

0 on success, otherwise one of the following error codes
NSM_VLAN_ERR_IFP_NOT_BOUND
NSM_PVLAN_ERR_INVALID_MODE
NSM_VLAN_ERR_VLAN_NOT_FOUND
NSM_PVLAN_ERR_NOT_CONFIGURED
NSM_PVLAN_ERR_NOT_PRIMARY_VLAN

4.2 smi_pvlan_msg.h File Reference

Defines data structures used by Private VLAN SMI APIs. `#include "pal.h"`

```
#include "message.h"
#include "thread.h"
#include "network.h"
#include "log.h"
#include "tlv.h"
#include "syslog.h"
#include <sys/types.h>
#include "pal_types.h"
#include "pal_socket.h"
#include "prefix.h"
```

Data Structures

- struct pSecondaryVlan
- struct pvlanPrimList
- struct pvlanSeconList
- struct pvlan_msg_

Defines

- #define **SMI_PVLAN_RANGE_MIN** 2
- #define **SMI_PVLAN_RANGE_MAX** 4094
- #define **SMI_PVLAN_CTYPE_VR_ID** 0
- #define **SMI_PVLAN_CTYPE_BR_ID** 1
- #define **SMI_PVLAN_CTYPE_VLAN_ID** 2
- #define **SMI_PVLAN_CTYPE_PVLAN_TYPE** 3
- #define **SMI_PVLAN_CTYPE_PVID** 4
- #define **SMI_PVLAN_CTYPE_IFNAME** 5
- #define **SMI_PVLAN_CTYPE_MODE** 6
- #define **SMI_PVLAN_CTYPE_PVID_BMP** 7
- #define **SMI_PVLAN_CTYPE_BR_NAME** 8
- #define **SMI_PVLAN_CTYPE_PRIM_VID** 9
- #define **SMI_PVLAN_CTYPE_PRIM_LIST** 10
- #define **SMI_PVLAN_CTYPE_SECON_LIST** 11
- #define **SMI_PVLAN_CTYPE_EXTENDED_1** 31

Typedefs

- typedef struct pvlan_msg_ **pvlan_msg**

Enumerations

- enum **smi_pvlan_port_mode** { **SMI_PVLAN_PORT_MODE_INVALID**, **SMI_PVLAN_PORT_MODE_HOST**, **SMI_PVLAN_PORT_MODE_PROMISCUOUS** }
- enum **smi_pvlan_type** { **SMI_PVLAN_NONE**, **SMI_PVLAN_COMMUNITY**, **SMI_PVLAN_ISOLATED**, **SMI_PVLAN_PRIMARY** }
- enum **secondaryVlanType** { **NSM_PRIVATE_VLAN_NONE**, **NSM_PRIVATE_VLAN_COMMUNITY**, **NSM_PRIVATE_VLAN_ISOLATED**, **NSM_PRIVATE_VLAN_PRIMARY** }

Functions

- int **smi_parse_pvlan** (u_char **pnt, u_int16_t *size, struct smi_msg_header *header, void *arg, SMI_CALLBACK callback)
- int **smi_encode_pvlan** (u_char **pnt, u_int16_t *size, pvlan_msg *msg)
- int **smi_decode_pvlan** (u_char **pnt, u_int16_t *size, pvlan_msg *msg)

4.2.1 Detailed Description

Defines data structures used by Private VLAN SMI APIs.

Index